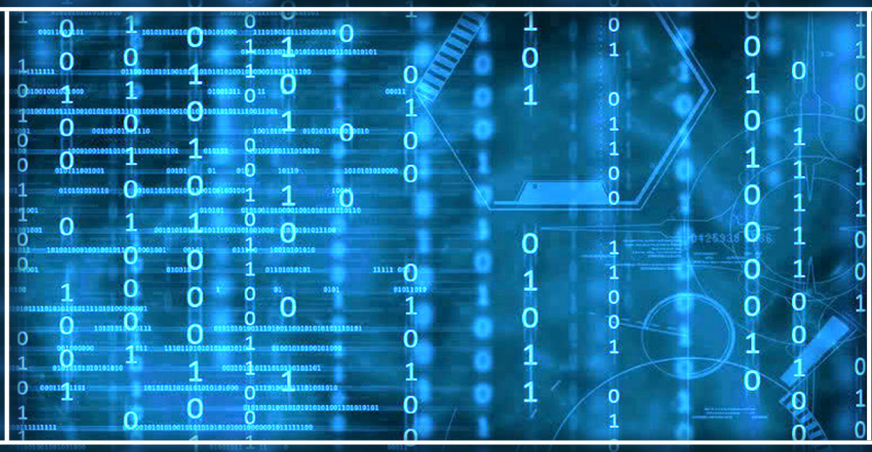
IJACSA
WHERE WISDOM SHARES
International Journal of Advanced Computer Science and Applications

Volume 16 Issue 9
September 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 9 September 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
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: Cognitive Biases: Understanding and Designing Fair AI Systems for Software Development
Authors: Sheriff Adepoju, Mildred Adepoju
PAGE 1 – 6

Paper 2: Multispectral Image Analysis Using Deep Neural Networks
Authors: Arun D. Kulkarni
PAGE 7 – 13

Paper 3: LegalSummNet: A Transformer-Based Model for Effective Legal Case Summarization
Authors: Md Farhad Kabir, Sohana Afrin Mitu, Sharmin Sultana, Belal Hossain, Rakibul Islam, Khandakar Rabbi Ahmed
PAGE 14 – 21

Paper 4: Real-Time Dynamic Pricing Using Machine Learning: Integrating Customer Sentiment and Predictive Models for E-Commerce
Authors: Areyfin Mohammed Yoshi, Arafat Rohan, Sohana Afrin Mitu, Md Masud Karim Rabbi, Shahanaj Akther, Khandakar Rabbi Ahmed
PAGE 22 – 30

Paper 5: A Scalable Microservices Architecture for Real-Time Data Processing in Cloud-Based Applications
Authors: Desidi Narsimha Reddy, Rahul Suryodai, Vinay Kumar S. B, M. Ambika, Elangovan Muniyandy, V. Rama Krishna, Bobonazarov Abdurasul
PAGE 31 – 42

Paper 6: A Graph-Based Deep Reinforcement Learning and Econometric Framework for Interpretable and Uncertainty-Aware Stablecoin Stability Assessment
Authors: Yaozhong Zhang, Quanrong Fang
PAGE 43 – 52

Paper 7: Embedded System for ECG Signal Monitoring and Fatigue Detection in Elderly Individuals Using Machine Learning Models
Authors: Chokri Baccouch, Chaima Bahar
PAGE 53 – 62

Paper 8: Deep Reinforcement Learning-Based Target Detection and Autonomous Obstacle Avoidance Control for UAV
Authors: Like Zhao, Hao Liu, Guangmin Gu, Fei Wan, Yanyang Feng
PAGE 63 – 78

Paper 9: Beyond Words: An Advanced Ensemble Framework for Unmasking AI-Generated Content Through Linguistic Fingerprinting
Authors: Ghada Y. Elwan, Doaa R. Fathy, Nahed M. El Desouky, Abeer S. Desuky
PAGE 79 – 91

Paper 10: Reliability Risk Assessment Approaches in Software Engineering: A Review Structured by Software Development LifeCycle (SDLC) Phases and Reliable Sub-Characteristics
Authors: Lehka Subramanium, Saadah Hassan, Mohd. Hafeez Osman, Hazura Zulzalil
PAGE 92 – 102

Paper 11: WiTS: A Wi-Fi-Based Human Action Recognition via Spatio-Temporal Hybrid Neural Network
Authors: Pengcheng Gao
PAGE 103 – 111

Paper 12: Automated Scoliosis Diagnosis in Spinal Imaging: Laboratory Validation, Clinical Limitations, and Systematic Implementation Challenge Review
Authors: Ervin Gubin Moung, Xie Aishu, Ali Farzamnia
PAGE 112 – 123

Paper 13: Adaptive Trust-Based Fault Tolerance for Multi-Drone Systems: Theory and Application in Agriculture
Authors: Atef GHARBI, Faheed A. F. Alrslani
PAGE 124 – 130

Paper 14: Method for Person Re-Identification with 2D-to-3D Image (Image-to-Video) Conversion
Authors: Kohei Arai
PAGE 131 – 138

Paper 15: Design of a Modular Architecture Based on AI and Blockchain for Personalized Microcredits Using Open Finance
Authors: Pedro Hidalgo, Ciro Rodríguez, Luis Bravo, Cesar Angulo
PAGE 139 – 153

Paper 16: Mathematical Representation of Netflow Analysis Decision Making Based on Production Logic
Authors: Alimdzhan Babadzhanov, Inomjon Yarashov, Maruf Juraev, Alisher Otakhonov, Adilbay Kudaybergenov, Rustam Utemuratov
PAGE 154 – 166

Paper 17: AI-Enabled Demand Forecasting, Technological Capability, and Supply Chain Performance: Empirical Evidence from the Global Logistics Sector
Authors: Mohamed Amine Frikha, Mariem Mrad
PAGE 167 – 174

Paper 18: A Starfish Optimization Algorithm-Based Federated Learning Approach for Financial Risk Prediction in Manufacturing Enterprises
Authors: Bin Liu, Liang Chen, Haitong Jiang, Rui Ma
PAGE 175 – 183

Paper 19: User Requirements of Adaptive Learning Through Digital Game-Based Learning: User-Centered Design Approach to Enhance the Language Literacy Development
Authors: Nur Atiqah Zaini, Tengku Siti Meriam Tengku Wook, Mohd Nor Akmal Khalid, Shahrul Azman Mohd Noah
PAGE 184 – 197

Paper 20: Towards More Effective Automatic Question Generation: A Hybrid Approach for Extracting Informative Sentences
Authors: Engy Yehia, Neama Hassan, Sayed AbdelGaber
PAGE 198 – 209

Paper 21: An Integrated Evaluation Using Enhanced Panel Factor Model and Machine Learning: Assessing the Level and Structure of Regional Coordinated Development in the Guangdong-Hong Kong-Macao Greater Bay Area
Authors: Li Shi, Ting Nie
PAGE 210 – 219

Paper 22: CrypTen-FL: A Secure Federated Learning Framework for Multi-Disease Prediction from MIMIC-IV Using Encrypted EHRs
Authors: Himanshu, Pushpendra Singh
PAGE 220 – 228

Paper 23: Is Metaverse Technology Ready to Welcome Online Banking Users?
Authors: Yantu Ma, Dalbir Singh, Siok Yee Tan, Meng Chun Lam, Ab Ghani Nur Laili, Haodong Guan, Fengjin Lei, Ahmad Sufril Azlan Mohamed
PAGE 229 – 239

Paper 24: Optimizing Image Retrieval: A Two-Step Content-Based Image Retrieval System Using Bag of Visual Words and Color Coherence Vectors
Authors: Muhammad Sauood, Muhammad Suzuri Hitam, Wan Nural Jawahir Hj Wan Yussof
PAGE 240 – 249

Paper 25: An Aggregated Dataset of Agile User Stories and Use Case Taxonomy for AI-Driven Research
Authors: Abdulrahim Alhaizaey, Majed Al-Mashari
PAGE 250 – 259

Paper 26: Comparative Evaluation of Centrality Measures for Detecting Significant Nodes in Social Networks
Authors: Hardeep Singh, Supreet Kaur, Karman Singh Sethi, Jai Sharma
PAGE 260 – 268

Paper 27: Neural Networks for Pest Diagnosis in Agriculture: A Global Literature Review
Authors: Heling Kristtel Masgo Ventura, Italo Maldonado Ramírez, Roberto Carlos Santa Cruz Acosta, Wilfredo Ruiz Camacho, Juan Eduardo Suarez Rivadeneira, José Celso Paredes Carranza, Mayra Pamela Musayón Díaz, Cesar R. Balcazar Zumaeta, Carlos Luis Lobatón Arenas, Juan Alberto Rojas Castillo, Eli Morales- Rojas
PAGE 269 – 284

Paper 28: Evaluating User Experience in a Public Sector Digital System Through Nielsen's Heuristic Approach
Authors: Nagyan Yosse Wibisono, Viany Utami Tjhin
PAGE 285 – 295

Paper 29: NEBULA Framework: An Adaptive Framework for Unstructured Description to Solve Cold Start Problem
Authors: I Gusti Agung Gede Arya Kadyanan, Ni Made Ary Esta Dewi Wirastuti, Gede Sukadarmika, Ngurah Agus Sanjaya ER
PAGE 296 – 305

Paper 30: Analysis of Factors Affecting Continuance Intention in Indonesian Digital Banks
Authors: Soros Lie, Viany Utami Tjhin
PAGE 306 – 319

Paper 31: Generating a Trading Strategy Using Candlestick Patterns with Machine Learning
Authors: Hussaina Bala Malami, Badamasi Imam Ya'u, Fatima Umar Zambuk, Mohannad Alkanan, Osman Elwasila, Mohammad Shuaib Mir, Mohammed Nasir Danmalam Bawa, Yonis Gulzar
PAGE 320 – 325

Paper 32: Advancing Speech Enhancement with Generative Adversarial Network-Autoencoder: A Robust Adversarial Autoencoder Approach
Authors: Mandar Diwakar, Brijendra Gupta
PAGE 326 – 332

Paper 33: Enhancing Smart City Safety: Deep Learning Approaches for Automatic Vehicle Accident Recognition
Authors: Ahad AlNemari, Shahad AlOtaibi, Majd Jada, Aeshah AlHarthi, Sara AlThuwaybi, Wojoud AlNemari, Nadan Marran, Abdulmajeed Alsufyani
PAGE 333 – 348

Paper 34: Prioritizing Non-Functional Requirements and Influencing Factors for API Quality Framework: An Industry Approach
Authors: Aumir Shabbir, Aziz Deraman, Mohamad Nor Bin Hassan, Kamal Uddin Sarker, Shahid Kamal
PAGE 349 – 361

Paper 35: Perceived Usefulness and Perceived Ease of Use as Predictors of Attitude Toward IoT Adoption Among Rice Farmers
Authors: Hermin Arrang, Sek Yong Wee, Nazrulazar Bin Bahaman, Jack Febrian Rusdi
PAGE 362 – 370

Paper 36: An Analytical Review of Environmental and Machine Learning Approaches in Dengue Prediction
Authors: Orlando Iparraguirre-Villanueva, Juan Chavez-Perez, Eddier Flores-Idrugo, Luis Chauca-Huete
PAGE 371 – 384

Paper 37: Assessing the Effectiveness of MCR-KSM for Waiting Waste Reduction: An Empirical Study
Authors: Nargis Fatima, Sumaira Nazir, Suriayati Chuprat
PAGE 385 – 394

Paper 38: A Review of Attention-Enhanced GRU Models with STL Decomposition for Food Loss Forecasting
Authors: Ru Poh Tan, Siew Mooi Lim, Kuan Yew Leong, Shee Chia Lee, Siaw Hong Liew, Jun Kit Chaw
PAGE 395 – 407

Paper 39: An Incremental LSTM Ensemble for Online Intrusion Detection in Software-Defined Networks
Authors: Raed Basfar, Mohamed Y. Dahab, Abdullah Marish Ali, Fathy Eassa, Kholoud Bajunaied
PAGE 408 – 419

Paper 40: Comprehensive Analysis of YOLOv8 + DeepSORT for Vehicle Tracking: HOTA and CLEAR-Based Evaluation
Authors: I Nyoman Eddy Indrayana, Made Sudarma, I Ketut Gede Darma Putra, Anak Agung Kompiang Oka Sudana
PAGE 420 – 432

Paper 41: Sentence-Level Indonesian Sign Language (BISINDO) Recognition Using 3D CNN-LSTM and 3D CNN-BiLSTM Models
Authors: Katriel Larissa Wiguna, Rojali
PAGE 433 – 443

Paper 42: Formal Verification of a Blockchain-Based Security Model for Personal Data Sharing Using the Dolev-Yao Model and ProVerif
Authors: Godwin Mandinyenya, Vusumuzi Malele
PAGE 444 – 456

Paper 43: Enhancing Cybersecurity Programs in Small and Medium Enterprises (SMEs): A Systematic Literature Review
Authors: Eliana Ludin, Masnizah Mohd, Fariza Fauzi
PAGE 457 – 464

Paper 44: Hybrid Fuzzy–PPO Control for Precision UAV Spraying
Authors: Ahmad B. Alkhodre, Adnan Ahmed Abi Sen, Yazed Alsaawy, Nour Mahmoud Bahbouh, ***Mohamed Benaida***
PAGE 465 – 474

Paper 45: Dynamic Assessment and Optimization Strategy for Brand Tourism Competitiveness in the Yangtze River Delta City Cluster Based on Entropy Weight-TOPSIS
Authors: Dongmei Wang, Daoyi Wu
PAGE 475 – 488

Paper 46: Edge-Integrated IoT and Computer Vision Framework for Real-Time Urban Flood Monitoring and Prediction
Authors: Rupesh Mandal, Bobby Sharma, Dibyajyoti Chutia
PAGE 489 – 503

Paper 47: Privacy-Aware ML Framework for Dynamic Query Formation in Multi-Dimensional Data
Authors: B Bhavani, Haritha Donavalli
PAGE 504 – 515

Paper 48: Performance Analysis of Spectrogram-Based Versus Raw Waveform-Based Deep Learning Models for Smoker Detection from Cough Audio
Authors: Widi Nugroho, Alhadi Bustamam, Rinaldi Anwar Buyung
PAGE 516 – 523

Paper 49: ProjectNavigator: A Software Project Management Approach Selection Assistant
Authors: Lin Dongzhi, Salfarina Abdullah
PAGE 524 – 531

Paper 50: A Hybrid RoBERTa-BiGRU-Attention Model for Accurate and Context-Aware Figurative Language Detection
Authors: Sreeja Balakrishnan, Rahul Suryodai, S. Manochitra, Jasgurpreet Singh Chohan, Karaka Ramakrishna Reddy, A. Smitha Kranthi, Ritu Sharma
PAGE 532 – 544

Paper 51: Enhancing Code Quality Through Automated Refactoring Using Transformer-Based Language Models
Authors: A. Sri Lakshmi, E. S. Sharmila Sigamany, Roopa Traisa, Raman Kumar, Karaka Ramakrishna Reddy, Jasgurpreet Singh Chohan, Aseel Smerat
PAGE 545 – 554

Paper 52: Real-Time Biomechanical Squat and Deadlift Posture Analysis Using Google Machine Learning Kit
Authors: Liew Yee Jie, Ting Tin Tin, Chaw Jun Kit, Ali Aitizaz, Ayodeji Olalekan Salau, Omolayo M. Ikumapayi, Lim Siew Mooi
PAGE 555 – 570

Paper 53: A Novel CNN-Based Feature Fusion Framework for Breast Cancer Ultrasound Image Classification
Authors: Mobarak Zourhri, Bouchaib Cherradi, Mohamed El Khaili
PAGE 571 – 580

Paper 54: Machine Learning for Recommender Systems Under Implicit Feedback and Class Imbalance
Authors: Younes KOULOU, Norelislam EL HAMI
PAGE 581 – 589

Paper 55: Mapping Elderly Residential Research During the Onset of Baby Boomer Aging: A Bibliometric Analysis
Authors: Keyi Xiao, Han Wang, Yan Ma
PAGE 590 – 599

Paper 56: Strategic Decision Support in Financial Management Using Deep Learning-Based Stock Price Prediction Models
Authors: Layth Almahadeen, Chinnapareddy Venkata Krishna Reddy, Roopa Traisa, Mukhamadiev Sanjar Isoevich, Lavanya Kongala, Janvi Anand Rathi, Revati Ramrao Rautrao
PAGE 600 – 610

Paper 57: Advancements in Texture Analysis and Classification: A Bibliometric Review of Entropy-Based Approaches
Authors: Muqaddas Abid, Muhammad Suzuri Hitam, Rozniza Ali, Muhammad Hammad
PAGE 611 – 619

Paper 58: A Review of Visualization Techniques for Duplicate Detection in Cancer Datasets
Authors: Nurul A. Emran, Ruhaila Maskat
PAGE 620 – 628

Paper 59: Hybrid Real Time Facial Emotions Recognition on Autistic Individuals
Authors: Fatima Ezzahrae El Rhatassi, Btihal El Ghali, Najima Daoudi
PAGE 629 – 635

Paper 60: Integrating YOLOv8 and IoT in a Computer Vision System for Child Detection in Smart Cities
Authors: Modhawi Alotaibi, Atheer Alruwaythi, Sara Alenazi, Maisaa Alsaedi
PAGE 636 – 644

Paper 61: D.M.A.I.H.: Deepfake-Inspired Few-Shot Learning Approach with Stable Diffusion for Digital Mourning
Authors: Btissam Acim, Hamid Ouhnni, Nassim Kharmoum, Soumia Ziti
PAGE 645 – 652

Paper 62: Re-engineering Grid-Based Quorum Replication into Binary Vote Assignment on Cloud: A Scalable Approach for Strong Consistency in Cloud Databases
Authors: Ainul Azila Che Fauzi, Noor Ashafiqa, Asiah Mat, Syerina Azlin Md Nasir, A. Noraziah
PAGE 653 – 659

Paper 63: Optimized Random Forest for High-Accuracy Autism Spectrum Disorder Detection via Phenotypic Data
Authors: Mohamed Gawish, Nada S. El-Askary, Mohamed Mabrouk Morsey, Abeer M. Mahmoud, Mostafa Aref, Taha Ibrahim El-Arif
PAGE 660 – 672

Paper 64: Robust Control of Cyber-Physical Teleoperation Systems for Synchronized Healthcare Supply Chain Management
Authors: Mariem Mrad, Mohamed Amine Frikha
PAGE 673 – 678

Paper 65: Sentiment Analysis Revisited: A Multi-Metric Comparative Study
Authors: Kamal Walji, Allae Erraissi, Abdelali ZAKRANI, Mouad Banane
PAGE 679 – 686

Paper 66: Enhanced Crow Search Algorithm with Cooperative Island Strategy for Energy-Aware Routing in Wireless Sensor Networks
Authors: Xiangqian LI, Xuemei ZHOU
PAGE 687 – 698

Paper 67: From Review to Practice: A Comparative Study and Decision-Support Framework for Sentiment Classification Models
Authors: Kamal Walji, Allae Erraissi, Abdelali ZAKRANI, Mouad Banane
PAGE 699 – 709

Paper 68: A Weighted Scoring Model of Heuristic-Based Workload Scheduling Approaches in Edge-Cloud Environments
Authors: Hasnae NOUHAS, Abdessamad BELANGOUR, Mahmoud NASSAR
PAGE 710 – 723

Paper 69: Optimization of Convolutional Neural Network Algorithm for Indonesian Sign Language Classification
Authors: Alvin Bintang Rebrastya, Sumarni Adi, Hanif Al Fatta, Windha Mega Pradnya Dhuhita, Ika Nur Fajri, Muhammad Hanafi
PAGE 724 – 733

Paper 70: Control System of Ocean Wave Simulator Using PID-Salp Swarm Algorithm
Authors: Affiani Machmudah, Juchen Li, Mahmud Iwan Solihin, Chiong Meng Choung, Wibowo Harso Nugroho, Ahmad Syafiul Mujahid, Sahlan, Abdul Ghofur
PAGE 734 – 742

Paper 71: Enhanced IoT Security Using Machine Learning Technology
Authors: Rawan Yousef Bukhowah, Alanoud Khaled Bu Dookhi, Mounir Frikha
PAGE 743 – 756

Paper 72: DAE-IDS: A Domain-Aware Ensemble Intrusion Detection System with Explainable AI for Industrial IoT Networks
Authors: Saifur Rahman
PAGE 757 – 767

Paper 73: Enhanced Fuzzy Clustering Approach for Overlapping Community Detection via Structural Neighborhood Similarity
Authors: Faiza Riaz Khawaja, Zuping Zhang, Abdul Hadi Riaz, Abdolraheem Khader, Ahmed Hamza Osman, Hani Moetque Aljahdali, Ali Ahmed
PAGE 768 – 778

Paper 74: Vision-Based Autonomous Localization of Fall Protection Anchor Points on Transmission Towers Using Multi-View Geometric Perception
Authors: Chunqing Yang, Yu Peng, Jian Yu, Dongfeng Yu, Rui Liu, Jiahui Chen
PAGE 779 – 788

Paper 75: A New Hybrid Approach Based on Discrete Wavelet Transform and Deep Learning for Traffic Sign Recognition in Autonomous Vehicles
Authors: Rim Trabelsi, Khaled Nouri
PAGE 789 – 798

Paper 76: A Computer-Aided Diagnosis System for Ulcerative Colitis Classification Using Vision Transformer
Authors: Dharmendra Gupta, Jayesh Gangrade, Yadvendra Pratap Singh, Shweta Gangrade
PAGE 799 – 805

Paper 77: Graph Neural Networks with Shapley-Value Explanations for Hierarchical Recommendation Systems
Authors: Redwane Nesmaoui, Mouad Louhichi, Mohamed Lazaar
PAGE 806 – 817

Paper 78: Benchmarking Large Language Models for Hate Speech Detection in Arabic Dialects: Focus on the Saudi Dialects
Authors: Omaima Fallatah
PAGE 818 – 825

Paper 79: Task-Oriented Evaluation of Assamese Tokenizers Using Sentiment Classification
Authors: Basab Nath, Sagar Tamang, Osman Elwasila, Yonis Gulzar
PAGE 826 – 836

Paper 80: Leveraging Distance-Optimized Transformers for High-Performance Arabic Short Answers Grading
Authors: Hatem M. Noaman, Mohsen Rashwan, Hazem Raafat
PAGE 837 – 845

Paper 81: A FOREX Trading System Based on Semi-Supervised News Classification, Market Sentiment Analysis, and GRU-CNN Deep Learning Models
Authors: Nabil MABROUK, Marouane CHIHAB, Younes CHIHAB
PAGE 846 – 859

Paper 82: Scalable Formal Verification of Modular Concurrent Systems: A Survey of Techniques, Tools and Challenges
Authors: Sawsen Khlifa, Chiheb Ameur Abid, Asma ben Letaifa, Belhassen Zouari
PAGE 860 – 880

Paper 83: New Explainable Overlapping Co-Clustering for Recommender Systems: Capturing Multifaceted Preferences with Enhanced Interpretability
Authors: Chiheb Eddine Ben Ncir, Mohammed Ibrahim Alattas
PAGE 881 – 894

Paper 84: Applying a Lightweight Graphics Library to Visually Corroborate Learning in Programming Introduction Courses
Authors: Claudia De La Fuente, Cristian Vidal-Silva, Liza Jeg´o-Mendoza, Patricia Pedrero-Valenzuela
PAGE 895 – 901

Paper 85: Mental Health Monitoring in Neurodivergent Children Using NeuroSky TGAM1: Real-Time EEG Signal Processing for Cognitive and Emotional Assessment
Authors: Erika Yolanda Aguilar Del Villar, Jes´us Jaime Moreno Escobar, Claudia Hern´andez Aguilar
PAGE 902 – 908

Paper 86: A Weakly Supervised MIL Approach to Fake News Detection via Propagation Tree Analysis
Authors: Shariq Bashir
PAGE 909 – 919

Paper 87: Hierarchical Adaptive Gap-Run TID Compression for Large-Scale Frequent Itemset Mining
Authors: Xin Dai, Chenjiao Liu, Xue Hao, Qichen Su
PAGE 920 – 926

Paper 88: Optimizing Energy Efficiency and Increasing Scalability in 6G-IoT Networks Through SDN, Duty Cycling, and AI-Driven Slicing
Authors: Marwah Albeladi, Kamal Jambi, Fathy E. Eassa, Maher Khemakhem
PAGE 927 – 941

Paper 89: Chaotic Compressed Sensing for Secure Image Transmission in LoRa IoT Systems
Authors: Chatchai Wannaboon, Shamsul Ammry Bin Shamsul Ridzwan, Sorawit Fong-In
PAGE 942 – 949

Paper 90: Predictive Models in Mental Health Based on Unsupervised Data Clustering
Authors: Inoc Rubio Paucar, Cesar Yactayo-Arias, Laberiano Andrade-Arenas
PAGE 950 – 962

Paper 91: TomDetLeaf: A Realistic Multi-Source Dataset for Real-Time Tomato Leaf Detection
Authors: Yassmine Ben Dhiab, Mohamed Ould-Elhassen Aoueileyine, Abdallah Namoun, Ridha Bouallegue
PAGE 963 – 970

Paper 92: Comparative Analysis of Statistical, Machine Learning, and Deep Learning Approaches for Frost Prediction in the Peruvian Altiplano
Authors: Fred Torres-Cruz, Dina Maribel Yana-Yucra, Richar Andre Vilca-Solorzano
PAGE 971 – 979

Paper 93: Towards the Hybrid Approach for Predicting Stroke Risk: A Feature Augmented Model
Authors: Ting Tin Tin, Wong Jia Qian, Ali Aitizaz, Ayodeji Olalekan Salau, Omolayo M. Ikumapayi, Sunday A. Afolalu
PAGE 980 – 1000

Paper 94: Socio-Technical Factors Influencing Business Intelligence Adoption in SMEs
Authors: Ibrahim Abdusalam Abubaker Alsibhawi, Hazura Binti Mohamed, Jamaiah Binti Yahaya
PAGE 1001 – 1016

Paper 95: An Improved BFT Algorithm in Traceability Data for Supply Chain
Authors: Zhiyong Liang, Rongwang Jiang, Ming Yang, Boxiong Yang
PAGE 1017 – 1022

Cognitive Biases: Understanding and Designing Fair AI Systems for Software Development

Sheriff Adepoju[1], Mildred Adepoju[2]
Department of Computer Science-College of Engineering, Prairie View A&M University, Texas, United States[1]
Department of Computer Information Systems-College of Engineering, Prairie View A&M University, Texas, United States[2]

***Abstract*—Artificial Intelligence (AI) systems increasingly influence decisions that affect people's lives, making fairness a core requirement. However, cognitive biases, systematic deviations in human judgment, can enter AI through data, modeling choices, and oversight, amplifying social inequities. This paper examines how three bias channels, data, algorithmic, and human, manifest across the software development lifecycle and synthesizes practical strategies for mitigation. Using a qualitative review of recent scholarship and real-world case studies, we distill a lightweight diagnostic framework that helps practitioners identify bias sources, evaluate mitigation options against effectiveness, feasibility, transparency, and scalability, and institutionalize routine audits. We illustrate the framework with representative vignettes and summarize trade-offs between fairness goals and model performance. Our analysis recommends diverse and well-documented datasets, fairness-aware learning and evaluation, third-party audits, and cross-functional collaboration as mutually reinforcing levers. The paper contributes a developer-oriented map of cognitive bias risks across data, model, and human processes, a four-criterion rubric for comparing mitigation techniques, and an actionable checklist that teams can embed in their pipelines. The results aim to support software and product teams in building AI systems that are both accurate and equitable.**

Keywords*—Cognitive biases; fair AI systems; algorithmic bias; software development; bias mitigation; fairness; software engineering

I. Introduction

AI systems process large volumes of data to support or automate high-stakes decisions. While these systems can outperform humans in speed and consistency, they are also vulnerable to cognitive biases, systematic deviations from rational judgment, initially characterized in psychology [1].

Well-documented failures show how bias can harm marginalized groups in production systems. For example, commercial face-analysis tools historically exhibited higher error rates for darker-skinned women due to under-representation in training data [2], and a widely used healthcare algorithm under-referred Black patients to high-risk care despite similar needs [3].

These failures illustrate that bias can enter via data, modeling decisions, and human oversight across the software development lifecycle (SDLC) [4]. To address such risks, researchers and regulators increasingly recommend third-party audits and process-level controls for accountable AI development and deployment [5], alongside broader socio-technical guidance for identifying and managing bias [6].

Despite a growing literature, practitioner-ready guidance that integrates data, modeling, and human-in-the-loop decisions into a cohesive, evaluable process remains fragmented [4], [7]–[9]. Teams need concise steps for diagnosing where cognitive biases enter and for selecting mitigation options under real-world constraints.

Research gap. Prior work richly characterizes data and algorithmic bias, but developer-oriented tools that connect bias identification to concrete mitigation choices and recurring audits across the SDLC are still limited.

Contributions. This paper (C1) maps cognitive-bias entry points across Data–Algorithmic–Human (D–A–H) layers; (C2) proposes a four-criterion rubric: effectiveness, feasibility, transparency, and scalability for comparing mitigation options; (C3) offers an actionable checklist and audit cadence; and (C4) illustrates usage with domain-neutral vignettes.

Paper organization: The remainder of the paper is organized as follows: Section II reviews related work across data, algorithmic, and human sources of bias. Section III describes the research design and the proposed D–A–H diagnostic framework with the four-criterion rubric. Section IV presents results from applying the framework and summarizes trade-offs. Section V discusses implications for software teams and governance. Section VI concludes with limitations and future work.

II. Related Work

Foundational psychology established that people rely on cognitive heuristics, anchoring, availability, and confirmation that systematically shape judgment and decision-making. These mechanisms explain how human-in-the-loop stages in AI pipelines can introduce bias and why oversight must be designed rather than assumed [1], [2].

Within data-driven systems, surveys synthesize how bias arises from sampling, labeling, target selection, and deployment context; they also review technical interventions for detection and mitigation [3]–[5]. Toolkits and libraries operate these interventions for practice (for example, fairness metrics and pre-/in-/post-processing mitigations) [6].

Governance and accountability work emphasizes that technical fixes must be paired with process controls, documentation, audits, role clarity, and decision logs to counter automation bias and institutional confirmation bias [7]–[9]. Socio-technical reviews further highlight sources and impacts of gender and social bias and outline process changes to address them [10], [11].

Empirical case studies demonstrate the stakes in production systems: face analysis systems have shown significant subgroup error gaps, healthcare resource allocation algorithms have under-referred Black patients, and criminal-justice risk scores have exhibited disparate error rates [12]–[14]. Sector-specific syntheses in imaging and practitioner studies on real-world adoption show progress and persistent practice gaps [15], [16].

Synthesis and gap. Prior literature provides what to do (techniques and guardrails) but remains fragmented across Data–Algorithmic–Human (D–A–H) layers. Teams still lack a concise, developer-oriented process that (i) pinpoints bias entry points across the SDLC, (ii) selects mitigations via a transparent rubric (effectiveness, feasibility, transparency, scalability), and (iii) institutionalizes routine audits. Our framework addresses this by packaging a D–A–H map, a four-criterion rubric, and an actionable checklist that converts the literature into a repeatable procedure (see Table I).

TABLE I REPRESENTATIVE LITERATURE ON BIAS ACROSS THE SDLC

Author(s)	Year	Key Findings	Biases Addressed
Tversky & Kahneman	1974	Defined cognitive biases as systematic deviations from rational judgment.	Anchoring bias, availability heuristic
Barocas et al.	2019	Explored how bias in training data affects AI outcomes.	Data bias, algorithmic bias
Mehrabi et al.	2021	Analyzed the various sources of bias in AI and proposed mitigation techniques.	Confirmation bias, selection bias
Obermeyer et al.	2019	Identified racial bias in healthcare algorithms and its impact on patient outcomes.	Racial bias, outcome bias
Mitchell et al.	2018	Suggested fairness-aware algorithms and bias auditing as key mitigation strategies.	Automation bias, data bias
Crawford	2021	Critiqued the social and ethical implications of AI bias.	Automation bias, confirmation bias

The 1974 Tversky-Kahneman research established the psychological roots of cognitive biases, which became important for studies about AI system bias production. The researchers at Barocas et al. [16] maintained their analysis within AI systems to demonstrate how prejudice in training datasets maintains institutional discriminatory practices. The research by Mehrabi et al. [4] created a classification structure for biases and an approach for reducing their occurrence.

The work of Obermeyer et al. [3] shows how healthcare algorithms using racial bias produce health inequality results. The practical consequences of defective AI systems on disadvantaged populations become apparent through their findings. The authors Mitchell et al. [15], along with Crawford [21], proposed fairness-aware algorithms and stringent bias auditing as practical methods to address bias. However, Crawford stressed the requirement for increased AI development accountability. Scientific research dismisses the need for immediate action on cognitive bias detection in AI to develop ethical solutions for technology.

III. METHODOLOGY

Our research design is summarized in Fig. 1. The main bias pathways and mitigation levers are depicted in Fig. 2. This section outlines the systematic approach to analyzing cognitive biases in AI systems and the strategies for designing fair and equitable AI models. The methodology combines qualitative research methods with a comprehensive framework for identifying, evaluating, and mitigating biases. By employing a multi-dimensional approach, this study ensures a thorough investigation of how cognitive biases manifest in AI systems and how to reduce their impact effectively.

A. Research Design

The research methodology uses qualitative methods to extensively evaluate literature observations and real-world case studies from authentic sources. Because of its suitability, the chosen research design effectively explores detailed topics at the intersection of cognitive biases and AI systems [17]. By merging theoretical information with practical examples, the research reveals standard patterns in bias appearance and methods for its mitigation.

The research design contains three primary stages that follow one another.

1) Literature review: Analyzing scholarly articles, technical reports, and industry guidelines on cognitive biases and their influence on AI systems. During this stage, researchers explore available knowledge about existing gaps and established mitigation methods [16], [4].

2) The project examines documented AI system studies from different sectors, including healthcare, criminal justice, and recruitment, to determine how biases manifest and what mitigation approaches are utilized, based on [3] and [5].

3) The proposed framework creates a system for classifying biases and strategy-assessment capabilities. A methodical approach enables an organized assessment of AI model susceptibility to biases and their effective correction [15].

B. Data Collection

This investigation used secondary data from conferences, peer-reviewed journals, technical reports, and case studies. We established the following conditions as we navigated our choice of resource materials:

- Relevance to cognitive biases in AI and software development.
- Empirical evidence of bias impact and mitigation outcomes [20].
- Inclusion of recent publications from the past ten years to capture modern breakthroughs and developing challenges.

The research relies on primary cognitive-bias literature from Tversky and Kahneman [1] together with present-day AI fairness examinations by Barocas et al. [16] and Mehrabi et al.

[4]. Practical industry reports about algorithm audits and bias assessment are included via Raji et al. [5].

C. Bias Identification Framework

The identification of cognitive biases in AI systems followed a systematic approach to classification as well as analysis. The framework groups cognitive biases into three main sections.

- The classification of data bias occurs when there are insufficient or improperly organized datasets [2].
- The fundamental process of model optimization and design selection at any stage causes bias to be referred to as Algorithmic Bias [4].
- Human bias originates from the subjective choices developers and end-users make [21].

This classification system enables researchers to analyze how biases develop throughout the AI development process and their effects on system results.

D. Evaluation Criteria for Bias Mitigation Strategies

The evaluation of bias mitigation strategies happened through an assessment of four vital components:

- A strategy proves effective when it decreases or eliminates biases from Artificial Intelligence output.
- The practicality of deploying the strategy throughout the Artificial Intelligence development pattern defines feasibility.
- Internal and external researchers must have accessible insight into the methods used to mitigate biases and understand them clearly throughout the process.
- The strategy can scale across different AI applications and domains according to its scalability measures [15].

The set criteria help professionals achieve an equitable rating of bias reduction strategies and their associated implementation difficulties.

E. Bias Mitigation Techniques

The research provides an analysis of chosen bias mitigation methods from four specified categories:

- A comprehensive data collection strategy employs diverse participants from target groups to reduce potential bias, according to [2].
- A training approach for fair algorithms integrates explicit fairness rules that prevent bias production [4].
- Regular audits through the AI lifecycle for identifying and fixing hidden biases happen through Bias Audits and Fairness Assessments [5].
- The collaborative approach brings together experts from developer roles with ethicists and social scientists to resolve team and human cognitive biases by using methods described by [21].

These methods' effectiveness, obstacles, and practical compatibility are explicitly evaluated for their role in real-world AI systems.

F. Data Analysis Approach

The study analyzes recurring themes through thematic analysis to examine bias manifestation patterns and methods to reduce them. The authors first coded the data from literature investigations and case studies before categorizing them following the bias identification framework benchmark criteria. This method enables researchers to understand the origin of cognitive bias and evaluate various intervention methods [18]. The assessment spans three different stages.

- Researchers analyzed the key data about bias varieties, their effects, and defense approaches within the obtained materials.
- The analysis unites coded data into designated themes, including data-related bias alongside algorithmic Bias and human-related Bias, and it detects new patterns that arise.
- This section unites the thematic information into a unified explanation about cognitive bias understanding and management in AI systems.

G. Validity and Reliability

The research team takes several measures to achieve both valid and reliable results.

The study implements data triangulation by combining academic literature analysis with case study results and industry report content to find confirmation. Industry experts in AI ethics and software development conduct a peer evaluation of the developed analysis framework and its conclusions. This method includes maintaining a detailed record of data origins, programming choices, and analytical processes through an Audit Trail system, improving research repeatability.

H. Ethical Considerations

The study implements ethical research procedures by adopting methods to show data accurately while reducing interpretive errors. Sources must be referenced accurately because the research team respects intellectual property rights. Analytical processes should be completely documented to achieve transparency throughout the work.

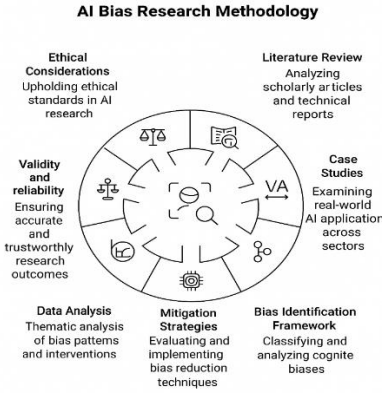


Fig. 1. AI bias research on methodology and ethical framework.

IV. Results

The findings of this study reveal the pervasive influence of cognitive biases on AI systems and the effectiveness of various mitigation strategies. Through the analysis of existing literature and case studies, the results highlight how cognitive biases manifest in AI development, the challenges associated with addressing these biases, and the impact of mitigation techniques in promoting fair and equitable AI systems. This section presents the key outcomes of the study, supported by empirical evidence and a comprehensive evaluation framework.

A. Manifestation of Cognitive Biases in AI Systems

The study identifies three primary channels through which cognitive biases infiltrate AI systems:

a) Data bias: Bias introduced through unrepresentative or incomplete datasets. For example, research by [2] shows that facial recognition systems exhibit higher error rates for darker-skinned individuals due to the lack of diverse training data. This bias can result in discriminatory outcomes in applications like law enforcement and hiring systems.

b) Algorithmic bias: Bias that emerges from the design and structure of AI models. Algorithms trained on biased data or optimized for specific performance metrics without fairness constraints may produce systematically biased outputs [4]. For instance, risk assessment algorithms in criminal justice have been shown to overestimate recidivism rates for minority groups [12].

c) Human bias: Bias stemming from the subjective decisions of AI developers and users. Confirmation bias, for example, may cause developers to prioritize data that aligns with their expectations, while automation bias leads users to trust AI outputs without critical examination [21].

B. Effectiveness of Bias Mitigation Strategies

The evaluation of mitigation strategies suggests that a multi-faceted approach is necessary to address cognitive biases effectively (Table II). The following techniques emerged as the most impactful:

a) Diverse and representative data collection: Ensuring inclusive datasets reduces the likelihood of biased outcomes. This study confirms that data diversity enhances model fairness [15].

b) Fairness-aware algorithms: Implementing fairness constraints during model training can mitigate algorithmic bias. For example, re-weighting data or applying adversarial debiasing has improved outcome parity across demographic groups [4].

c) Bias audits and fairness assessments: Regular audits help identify and address hidden biases. Prior work advocates for third-party audits to enhance accountability and transparency in AI deployment [5].

TABLE II Mitigation Strategies Evaluated against Effectiveness, Feasibility, Transparency, and Scalability

Bias Mitigation Strategy	Effectiveness	Feasibility	Transparency	Scalability
Diverse and Representative Data	High – Reduces data-driven biases (Mitchell et al., 2018)	Moderate – Requires significant data collection and curation	High – Transparent if dataset characteristics are disclosed	Moderate – Feasible with appropriate resources and policies
Fairness-Aware Algorithms	High – Mitigates algorithmic disparities (Mehrabi et al., 2021)	Moderate – Involves additional algorithm design complexity	Moderate – Depends on documentation of model adjustments	Low – Requires case-by-case customization
Bias Audits and Fairness Checks	Moderate – Identifies hidden biases (Raji et al., 2020)	High – Practical with audit tools and protocols	High – Facilitates external and internal review processes	High – Applicable across diverse AI systems

C. Challenges in Implementing Bias Mitigation

Despite the effectiveness of these strategies, several challenges limit their implementation:

- The process of making data fairer usually leads to reduced model precision. High-value domains, including healthcare and criminal justice, experience the most intense difficulties between fairness improvement and prediction accuracy [19].
- Low budgetary funds prevent smaller companies from executing complete audits and fairness-aware algorithm deployments [16].
- Eliminating human biases inherent in AI decision-making systems becomes harder due to their complexity. Applying cultural and organizational changes is essential to handle this matter [21].

D. Impact of Bias Mitigation on AI Outcomes

When AI models use bias mitigation frameworks, they generate results that minimize inequities—implementing healthcare algorithms using demographic parity adjustments eliminated disparities in patient treatment order (Obermeyer et al., 2019). Predictive accuracy maintained its high level while fairness-aware models achieved better hiring equity in their results [15].

Integrating technical methods alongside organizational measures and ethical guidelines helps AI developers reduce harmful cognitive biases, leading to better trust in AI systems. A complete approach enables AI system developers to construct efficient technology that delivers accuracy, fairness, and social responsibility.

V. Discussion

The research demonstrates how cognitive biases strongly affect AI systems operating in software development while identifying successful mitigation techniques. This section analyzes the results obtained by explaining their meaning and connecting them to previous field research.

A. The Influence of Cognitive Biases on AI Systems

Different stages throughout the operation of AI systems enable cognitive biases to affect data collection, algorithmic processing, and human oversight. AI models developed using datasets tend to reproduce the biases present in societal and structural inequalities because their training relies on these datasets [16]. AI systems develop biased outputs because developers tend to choose information confirming their initial beliefs, reinforcing faulty preconceptions [21]. Algorithmic bias creates severe problems during hiring and criminal justice operations since biased systems contribute to maintaining social disparities [4].

Obermeyer et al. [3] matched this study's findings about bias in algorithms, which are primarily caused by sparse datasets. The underlying bias within facial recognition technology produces higher identification errors among population groups because training algorithms adopt prejudiced information [2]. The research validates Mitchell et al. [15] position by showing that data quality combined with diversity serves as a solution for bias prevention.

B. Effectiveness of Bias Mitigation Strategies

The evaluation demonstrates that multiple strategies must be employed to minimize bias since any single approach provides inadequate elimination results. The collection of diverse representative data proved to be a compelling strategy for managing data bias according to Mehrabi et al. [4]. Data diversity ensures the collection of diverse populations and contextual characteristics, which minimizes the occurrence of exclusionary outcomes according to Mitchell et al. [15]. According to previous research, obtaining diverse datasets faces hindrances because of privacy restrictions and resource scarcity [16].

Integrating fairness constraints during model training enables fair machine learning algorithms, according to Mehrabi et al. [4]. The investigators at Obermeyer et al. [3] demonstrated that healthcare applications achieved higher fairness levels through data training methods that adjusted weights for balancing demographic characteristics. Deploying fair models demands strategic adjustment because doing so might decrease predictive capability [19].

According to Mitchell et al., the process of conducting bias audits and fairness assessments makes it possible to detect and resolve secretive biases [15]. Sending models for regular audits enables the detection of unfairness through enhanced transparency. The authors support Raji et al. [5] in recommending independent third-party auditing procedures to maintain objectivity and accountability within AI systems.

C. Challenges in Bias Mitigation

While these strategies achieve their intended goals, substantial difficulties remain in the present situation. The key drawback emerges from the compromise between achieving fair goals and maintaining model performance quality. According to Corbett-Davies & Goel [19], implementing fairness constraints adversely affects model performance, especially when datasets are unbalanced. The need for simultaneous accuracy and fairness poses significant challenges in high-consequence sectors such as healthcare and criminal justice systems.

The challenges stem from human involvement, which results in biased outputs. The encoding process of developer biases occurs when developers make choices about data selection during AI system design [21]. The effectiveness of bias awareness training is limited by cultural and organizational resistance to change, but it allows critical reflection to reduce biases [15].

D. Implications for AI System Design

The study findings establish a critical value for using multiple strategies to fight bias in computing systems. The complete lifecycle development of AI demands that software developers and stakeholders maintain transparency and accountability, according to Raji et al. [5]. Testing systems for fairness should be performed at different points during the AI development process, from data gathering to building the model and post-deployment assessment, to guarantee equitable results [4].

Resolving social and ethical dimensions in AI bias requires interdisciplinary cooperation between computer scientists, ethicists, and social scientists [21]. Interdisciplinary joint efforts between experts provide comprehensive perspectives and guidance to manage the challenges that emerge from technical enhancements versus social fairness goals [16].

E. Future Research Directions

Refined research is necessary to establish scalable automatic bias reduction methods, which should be integrated into current AI operational procedures [5]. Investigating bias audit effects on fairness-aware algorithms over time should be a future research goal, and evaluating their sustained equitable outcomes [15]. The analysis of new AI applications demands additional research because they introduce novel sources of bias, including generative models and autonomous decision systems [2].

The correct approach to handling AI system cognitive biases involves integrating technical solutions, organizational frameworks, and ethical considerations. AI system developers can generate fairer and more transparent system designs by acknowledging and addressing these biases according to Mehrabi et al. [4].

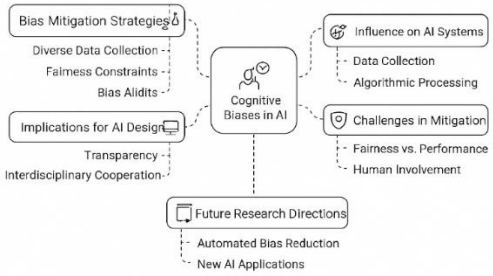


Fig. 2. Cognitive biases in AI: Influence, migration, and future directions.

VI. Conclusion

This paper provides a developer-oriented map of bias entry points across Data–Algorithmic–Human layers, a four-criterion rubric (effectiveness, feasibility, transparency, scalability) for comparing mitigation options, and an actionable checklist with audit cadence. Applied to representative vignettes, the framework makes trade-offs explicit and helps teams select interventions that fit product and process constraints. For software organizations, the immediate next steps are to (i) require dataset documentation and subgroup evaluation in CI/CD, (ii) adopt fairness-aware training and testing where harms are plausible, and (iii) schedule internal or third-party audits. Future work will validate the checklist prospectively in industry settings and extend the rubric to emerging generative AI risks.

References

[1] A. Tversky and D. Kahneman, "Judgment under uncertainty: Heuristics and biases," Science, vol. 185, no. 4157, pp. 1124–1131, 1974.

[2] J. Buolamwini and T. Gebru, "Gender shades: Intersectional accuracy disparities in commercial gender classification," in Proc. Conf. Fairness, Accountability, and Transparency (FAT*), PMLR, pp. 77–91, 2018.

[3] Z. Obermeyer, B. Powers, C. Vogeli, and S. Mullainathan, "Dissecting racial bias in an algorithm used to manage the health of populations," Science, vol. 366, no. 6464, pp. 447–453, 2019.

[4] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," ACM Computing Surveys, vol. 54, no. 6, 2021.

[5] I. D. Raji et al., "Closing the AI accountability gap: Defining auditing and its role in AI governance," in Proc. ACM Conf. Fairness, Accountability, and Transparency (FAT*), 2020

[6] R. Schwartz, A. Vassilev, K. Greene, L. Perine, A. Burt, and P. Hall, Towards a Standard for Identifying and Managing Bias in Artificial Intelligence. Gaithersburg, MD: National Institute of Standards and Technology, 2022.

[7] N. Kordzadeh and M. Ghasemaghaei, "Algorithmic bias: Review, synthesis, and future research directions," European Journal of Information Systems, vol. 31, no. 3, pp. 388–409, 2022.

[8] A. Balayn, C. Lofi, and G. J. Houben, "Managing bias and unfairness in data for decision support: A survey of machine learning and data engineering approaches," The VLDB Journal, vol. 30, no. 5, pp. 739–768, 2021.

[9] D. Varona and J. L. Suárez, "Discrimination, bias, fairness, and trustworthy AI," Applied Sciences, vol. 12, no. 12, p. 5826, 2022.

[10] P. Hall and D. Ellis, "A systematic review of socio-technical gender bias in AI algorithms," Online Information Review, vol. 47, no. 7, pp. 1264–1279, 2023.

[11] D. Cirillo and M. J. Rementeria, "Bias and fairness in machine learning and artificial intelligence," in Sex and Gender Bias in Technology and Artificial Intelligence, Academic Press, pp. 57–75, 2022.

[12] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias," ProPublica, 2016.

[13] A. S. Tejani, Y. S. Ng, Y. Xi, and J. C. Rayan, "Understanding and mitigating bias in imaging artificial intelligence," Radiographics, vol. 44, no. 5, e230067, 2024.

[14] A. Pant, R. Hoda, C. Tantithamthavorn, and B. Turhan, "Navigating fairness: Practitioners' understanding, challenges, and strategies in AI/ML development," arXiv:2403.15481, 2024.

[15] S. Mitchell, E. Potash, S. Barocas, A. D'Amour, and K. Lum, "Prediction-based decisions and fairness: A catalogue of choices," in Proc. ACM FAT*, 2018.

[16] S. Barocas, M. Hardt, and A. Narayanan, Fairness and Machine Learning. fairmlbook.org, 2019.

[17] J. W. Creswell, Research Design: Qualitative, Quantitative, and Mixed Methods Approaches, 4th ed. Thousand Oaks, CA: Sage, 2014.

[18] V. Braun and V. Clarke, "Using thematic analysis in psychology," Qualitative Research in Psychology, vol. 3, no. 2, pp. 77–101, 2006.

[19] S. Corbett-Davies and S. Goel, "The measure and mismeasure of fairness: A critical review of fair machine learning," arXiv:1808.00023, 2018

[20] R. K. E. Bellamy et al., "AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias," IBM Journal of Research and Development, vol. 63, no. 4/5, 2019.

[21] K. Crawford, Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence. New Haven, CT, USA: Yale Univ. Press, 2021.

Multispectral Image Analysis Using Deep Neural Networks

Arun D. Kulkarni
Computer Science Department, The University of Texas, Tyler, TX 75799, USA

***Abstract*—Multispectral image classification plays a crucial role in remote sensing applications such as land cover mapping, agricultural monitoring, and environmental surveillance. Traditional classification techniques, including the Maximum Likelihood Classifier (MLC), Support Vector Machine (SVM), Decision Tree (DT), and Multi-Layer Perceptron (MLP), often struggle with the complexity and high dimensionality of multispectral data. Recent advances in deep learning have revolutionized the field of remote sensing by enabling the extraction of high-level, abstract features from raw input data. In this paper, we explore the application of Deep Neural Networks (DNNs) for pixel-wise classification in multispectral imagery. DNNs are capable of learning informative and hierarchical representations, which have demonstrated significant success in a wide range of computer vision tasks. We propose and implement a simple DNN architecture consisting of six layers: an input layer (representing reflectance values across spectral bands), a fully connected layer, a batch normalization layer, a ReLU activation layer, another fully connected layer, and a final SoftMax output layer for classification. Each pixel is represented by a vector of spectral reflectance values. We evaluated our model using two Landsat scenes, one from the New Orleans area and the other from the Mississippi River bottomland area. The proposed DNN achieved classification accuracy of 97.44% and 95.74%, respectively, on these datasets, demonstrating the effectiveness of deep learning for multispectral image classification.**

***Keywords*—Remote sensing; classification; deep neural networks; Landsat scene**

I. Introduction

Multispectral imaging has become an essential tool in the field of remote sensing, enabling the acquisition of detailed information about the Earth's surface by capturing data across multiple distinct spectral bands. Unlike conventional RGB imagery, which captures only three visible bands (red, green, and blue), multispectral images span a broader range of the electromagnetic spectrum, including the visible, near-infrared (NIR), and shortwave infrared (SWIR) regions, providing richer spectral detail. This additional spectral information enhances the ability to distinguish between materials and land cover types, making multispectral imagery invaluable for a variety of applications, such as agricultural monitoring, land use classification, forestry, environmental change detection, and disaster assessment.

In remote sensing, pixel-based classification is one of the preferred methods for analyzing multispectral images. However, classifying multispectral images presents several challenges. The increased dimensionality of the data, combined with high inter-class similarity and intra-class variability, makes it difficult for traditional machine learning (ML) algorithms to achieve high classification accuracy. Common ML techniques used for pixel-based classification of multispectral images include the maximum likelihood classifier (MLC), decision tree (DT), support vector machine (SVM), random forest (RF), k-nearest neighbors (k-NN), and neural network (NN) models. In pixel-based classification, each pixel is represented as a vector of reflectance values across multiple spectral bands. A few sample pixels are selected from homogeneous areas representing distinct ground categories to form the training dataset. The classifier model is trained using this data, and once trained, the model is applied to classify the entire image scene. In the late 1980s, neural network (NN) models such as the multi-layer perceptron model with backpropagation learning were used for pixel-based classification of multispectral imagery. One advantage of NN models is that they are more efficient compared to conventional machine learning models, such as maximum likelihood classification (MLC).

In recent years, deep learning has revolutionized the field of image analysis by introducing models capable of learning complex, non-linear representations of data directly from raw inputs. A major limitation of early NN models using gradient descent with sigmoid activation functions was the vanishing gradient problem, where saturation of activation functions led to slow convergence during training. To mitigate this issue, deep learning (DL) models often employ entropy-based loss functions along with rectified linear unit (ReLU) activations in the output layer. Overfitting commonly occurs when the dataset is small. To address this, various regularization techniques such as dropout and bagging are employed. DL models can be trained on large datasets and can achieve high classification accuracy. DL-based pixel classification for multispectral images involves designing architectures that support pixel-wise data representation and classification. By adopting DL techniques, it becomes possible to extract more abstract and robust feature representations, thereby improving classification performance. DL models can learn informative representations of raw input data through multiple levels of abstraction. These features have achieved notable success across many computer vision tasks. Among deep learning architectures, convolutional neural networks (CNNs) have demonstrated exceptional performance in tasks such as image classification, object detection, and segmentation. In the context of multispectral image classification, deep learning offers several key advantages: it eliminates the need for manual feature extraction, captures both spectral and spatial dependencies, and scales effectively with large and high-dimensional datasets. Furthermore, advanced architectures such as 3D CNNs, recurrent neural networks (RNNs), and hybrid models have been proposed to better exploit

the unique characteristics of multispectral and hyperspectral data. Transfer learning and data augmentation techniques have also been utilized to address the common challenge of limited labeled datasets in remote sensing. These innovations have led to significant improvements in classification performance, establishing deep learning as a leading approach in this domain.

This study focuses on the classification of multispectral images using deep learning techniques. We propose a simple one-dimensional deep neural network (DNN) architecture to classify pixels in multispectral imagery. To validate our approach, we implemented a six-layer DNN using the MATLAB Deep Learning Toolbox. We analyzed two Landsat scenes: one from the New Orleans area and another from the Mississippi River bottomland area. Training data were extracted by displaying the scenes on a monitor and selecting small homogeneous regions representing distinct ground categories. For the Mississippi scene, four training areas were selected, each corresponding to a different category. For the New Orleans scene, three training areas were selected, each representing a distinct category. The remainder of the paper is organized as follows: Section II reviews related work; Section III describes the proposed approach; Section IV presents the experiments and results; and Section V concludes the study.

II. Related Work

Earlier well-known machine learning (ML) algorithms, such as the maximum likelihood classifier (MLC), support vector machine (SVM), decision tree (DT), random forest (RF) and k-Nearest Neighbor (k-NN) were used for classifying pixels in multispectral images. The MLC algorithm is one of the most well-known algorithms. It assumes the normal distribution for reflectance values and calculates the mean vector and covariance matrix for each class using training data. The classifier uses Bayes' rule to calculate posterior probabilities and assigns a pixel to the class with the highest posterior probability. The SVM is a binary classifier that assigns a sample to one of the two linearly separable classes. In the SVM algorithm, two hyper-planes are selected so as not only to maximize the distance between the two classes but also not to include any points between them. The SVM algorithm is extended to nonlinearly separable classes by mapping samples to a higher-dimensional feature space. The SVM algorithm is appealing for Landsat data analysis because of its ability to successfully handle small datasets, often producing higher classification accuracy than traditional methods. Moumtrakis et al. [1] have provided a review of usage of SVM in remote sensing. Kulkarni and Shrestha [2] have used DT to classify Landsat imagery. Neural networks are preferred for classification because of their parallel processing capabilities as well as learning and decision-making abilities. Several studies aimed at evaluating the performance of neural networks in comparison with traditional statistical methods for remote sensing applications are available. The rapid uptake of neural approaches in remote sensing is due mainly to their widely demonstrated ability to: 1) perform more accurately than other techniques, such as statistical classifiers, particularly when the feature space is complex and the source data has diverse statistical distributions; 2) perform more rapidly than other techniques, such as different sensors; 3) incorporate a priori knowledge and realistic physical constraints into the analysis, incorporate different types of data (including those from different sensors) into the analysis, thus facilitating synergistic studies. Neural networks with learning algorithms such as backpropagation (BP) can learn from training samples and are used in Landsat data analysis. Fuzzy-neural systems that combine neural networks and fuzzy logic have been used to classify pixels in remote sensing imagery.

Deep learning has achieved impressive performance with applications such as image classification and object detection. Deep learning is the fastest growing trend in big data analysis. Deep learning is characterized by neural networks that usually have more than two hidden layers. Zhang, et al. [3] provide a tutorial on the use of deep learning in remote sensing. Remote sensing data are often multispectral and face big data challenges. Deep learning provides more methodologies to train deep NN architectures, such as autoencoder models Convolution Neural Networks (CNNs). Earlier well-known CNN models include Alex Net, VGG Net, and Reset. Feature representations learned by CNNs are highly effective in object recognition. Zhu et al. [4] provide an overview of deep learning techniques for remote sensing. They discuss various architectures, data sets, and performance metrics commonly used in multispectral image classification. Many comprehensive survey articles on deep learning applied to remote sensing are available in the literature [5-9]. CNNs have become instrumental in extracting spatial and spectral features from satellite and aerial imagery for a variety of applications. CNNs have demonstrated significant performance improvements in image classification, object detection, and semantic segmentation, making them highly suitable for remote sensing applications. In remote sensing, CNNs have been used for applications such as land cover and land use classification, object detection, semantic segmentation, change detection, and agriculture monitoring. Zhu et al. [4], in their review article, summarize the basic principles of deep learning and its research progress and typical applications in remote sensing. Their article focuses on the research status of deep learning in remote sensing image classification and object detection. Deep learning techniques, especially CNNs, have revolutionized the field of remote sensing, particularly pixel-based classification tasks. The DL approaches for pixel-based multispectral image classification involve 1) preparing an input vector representing a pixel in the scene, 2) developing and training a hierarchical DL model, and 3) classifying the entire scene. In the first step, the input vector could represent spectral features, spatial features, or spectral and spatial features. Hidden layers in the deep network structure are designed to learn feature representations of the input data. The last classification stage involves decision making. Utilizing learned features from previous stages. In hard classification, the classifier directly outputs an integer number as the label of each sample. Hu et al. [10] train a simple one-dimensional CNN that contains five layers: an input layer, a convolution layer, a max-pooling layer, a fully connected layer, and an output layer that was used to classify pixels in hyper-spectral imagery. Feature vectors representing the spectral signatures were used as an input. Their results demonstrate that their proposed method can achieve better classification performance compared with SVM-based and conventional deep NN-based classifiers. Maggiori et al. [11] proposed an end-to-end framework for the dense, pixelwise classification of satellite imagery with convolutional neural networks (CNNs). In their framework, CNNs are directly trained

to produce classification maps out of the input images. Kussul et al. [12] proposed a multi-level deep learning architecture that targets land cover and crop type classification from multitemporal and multisource satellite imagery. Their results show that architecture with an ensemble of CNNs outperforms the one with a multi-layer perception. Kulkarni [13] proposed an architecture to classify pixels based on their spectral signatures using CNN models. In the proposed framework, each feature vector was mapped to a two-dimensional image. The mapped images were then classified to respective classes. Li et al. [14] provide a survey of deep learning techniques for remote sensing image mapping. They elucidate four predominant strategies for matching: arar-based, feature-based, regression-based, and unsupervised learning. Lian et al. [15] provide an overview of advances in deep learning-based spatiotemporal fusion methods for remote sensing images.

U-Net is one of the types of deep network models that is specifically built for image segmentation, which was introduced in 2015 [16]. Long et al. [17] have shown that CNN trained end-to-end, pixel-to-pixel, exceed the state-of-the-art in semantic segmentation. They define and detail the space of fully convolutional networks and explain their application in spatially dense prediction tasks. Solórzano et al [18] use U-net for land cover classification. Sahin et al. [19] trained the U-Net with images from the cotyledon emergence through to the subsequent growth stages and evaluated images of crops in the last stage of growth. The results of their experiment demonstrate that the proposed method, with suitable input, can be used for crop and weed detection or segmentation. Wiratama et al. [20] proposed a change detection algorithm on multi-spectral images based on a feature-level U-net. Their algorithm gave a better performance compared to the existing change detection algorithms. Bai et al. [21] provide a review of deep learning methods for change detection in remote sensing images.

III. Proposed Approach

A. Data Collection and Pre-processing

In this study, we used two Landsat scenes obtained Operational Land Imager (OLI) representing the New Orleans area located at Latitude N 30.55580 and Longitude W 89.92440. The second scene represents the Mississippi River bottomland area located at Latitude N 34.65710 and Longitude W 90.40900. We used spectral bands 2, 3, 4, 5, and 7 as these bands showed the maximum variance. The training set data were obtained by displaying the scenes and selecting homogeneous areas representing the classes. The training set data were used to train the DNN models. For the New Orleans scene, the training set contained a total of 1200 pixels, 400 pixels, and for the Mississippi River bottomland scene, the total number of training data was 3600 pixels, 900 from each class. Each pixel was represented by a vector of size 5. Seventy percent random samples were used to train the DNN model, and 30 percent random samples were used for testing. The reflectance values were normalized between 0 and 10.

B. DNN Architecture

The classification model used in this study was a simple Deep Neural Network (DNN) built in MATLAB, comprising six layers. The network was designed to process the multispectral feature vectors and classify the pixels into distinct categories based on their spectral characteristics. The architecture of the model is shown in Fig. 1. The layers in the model are as below.

1) Feature input layer: The first layer of the network was the input layer, where each pixel's feature vector, consisting of five spectral reflectance values (one for each band), was fed into the network. This layer was designed to accept data with a dimensionality corresponding to the number of spectral bands used.

2) Fully connected layer: The second layer of the network was a fully connected (dense) layer, which applied a set of learnable weights to the input features. This layer allowed the network to begin learning the complex relationships between the spectral bands.

3) Batch normalization layer: To enhance the training stability and speed up convergence, a batch normalization layer was applied. This layer normalizes the output of the fully connected layer, ensuring that the data fed into the next layer is centered and scaled appropriately. This normalization helped reduce internal covariate shift and improved model generalization.

4) ReLU (Rectified Linear Unit) activation layer: The next layer was a ReLU activation layer. ReLU introduces non-linearity into the model, allowing it to learn more complex features from the input data. The ReLU function was applied elementwise to the outputs of the batch normalization layer, ensuring that only positive values passed through to subsequent layers. ReLU later helps mitigate the vanishing gradient problem

5) Fully connected layer: The fifth layer was another fully connected layer, which aimed to further abstract and combine features learned from the previous layers. This layer had a greater number of neurons than the first fully connected layer, enabling the model to capture more intricate relationships in the data.

6) SoftMax layer: The fully connected layer was followed by a SoftMax layer. SoftMax layer takes a vector of raw scores (often called logits) from the final hidden layer and converts them into a probability distribution over multiple classes. The scores are exponentiated and then normalized so that they sum to 1.

The model was trained using the training data collected from the homogeneous regions. During training, the model optimized its weights using backpropagation and stochastic gradient descent (SGD). The loss function used for training was categorical cross-entropy, as the goal was to classify each pixel into one of several predefined classes. A portion of the training data was set aside for validation to monitor the model's performance during training and to prevent overfitting. Training was performed for a predetermined number of epochs, with the learning rate adjusted dynamically to ensure efficient convergence. Early stopping was used to prevent overfitting, where training was halted once the validation accuracy plateaued over several epochs. To evaluate the performance of the trained model, several metrics were computed, including overall accuracy, precision, recall, F1-score, and ROC curves.

The predicted categories were compared to the true categories in test sets.

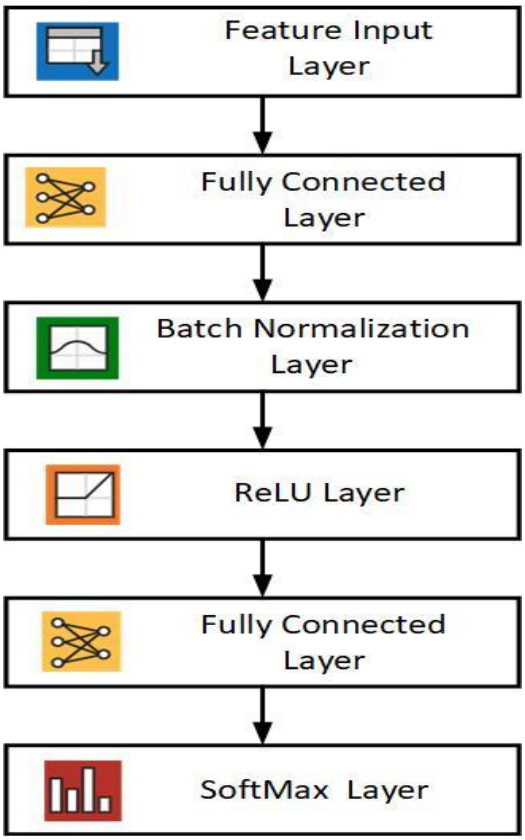


Fig. 1. Deep Neural Network (DNN) architecture.

IV. Implementation and Results

In this research work, we implemented the DNN model using MATLAB scripts from deep learning and analyzed two Landsat scenes.

A. Example 1: New Orleans Scene

The scene was obtained by Landsat-8 OLI, representing the New Orleans area. The path and row numbers for the scenes are 22 and 39, respectively. To generate the training data, we considered a scene of the size 512 by 512 pixels. Three small homogeneous areas were selected as training sets that represent three classes: water, land, and vegetation. The training data contains 1200 samples, 400 from each class. We selected bands 2, 3, 4, 5, and 7. The spectral signatures obtained from mean vectors of the classes are shown in Fig. 2. The scatter plot for the dataset is shown in Fig. 3. We used 70 percent randomly selected samples for training and 30 percent for validation. With the DNN, we obtained the overall accuracy of 97.5 percent. The learning progress curve for the DNN is shown in Fig. 4. The confusion matrix is shown in Fig. 5. The ROC curves and classified output are shown in Fig. 6 and 7. Table I shows the evaluation metric with Recall, Precision, F-Score, accuracy, and specificity for the classes in the data set.

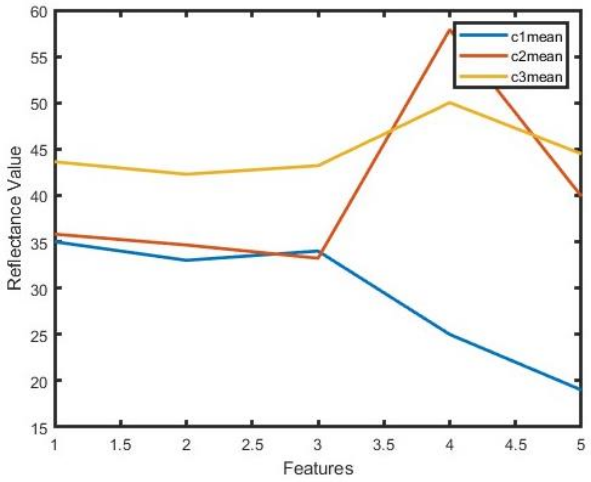


Fig. 2. Spectral signatures (New Orleans scene).

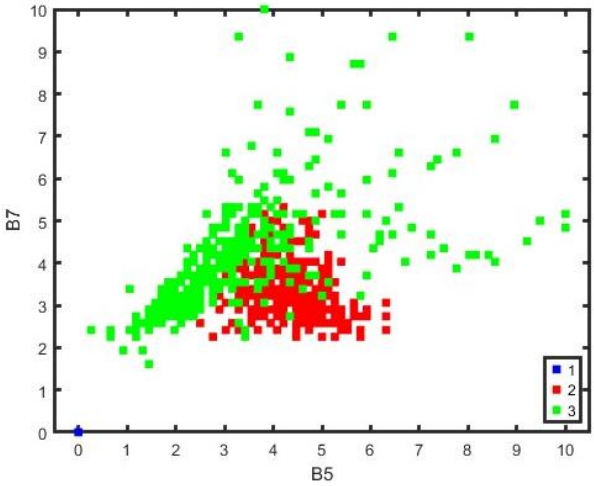


Fig. 3. Scatter plot (New Orleans scene).

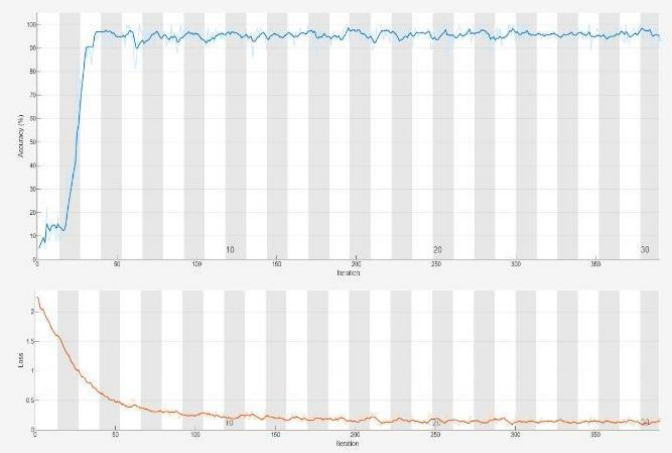


Fig. 4. Training progress (New Orleans scene).

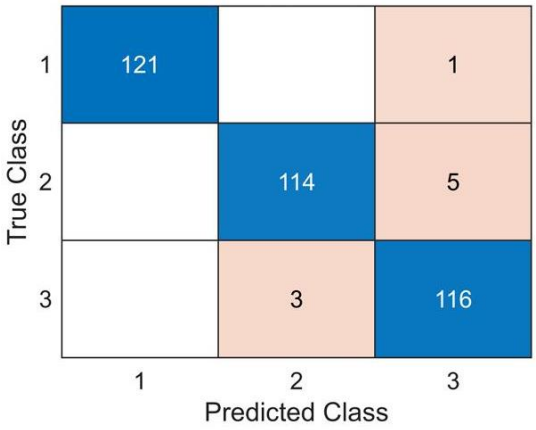


Fig. 5. Confusion matrix (New Orlean scene).

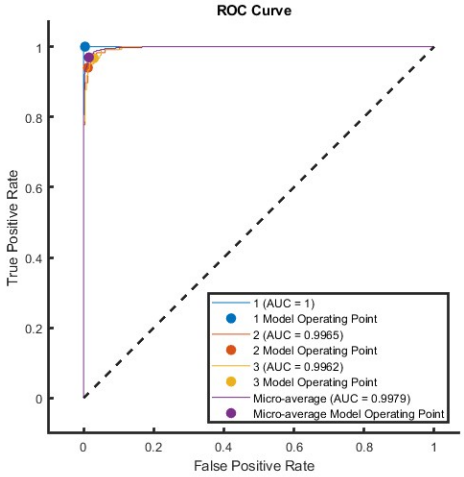


Fig. 6. ROC Curve (New Orleans scene).

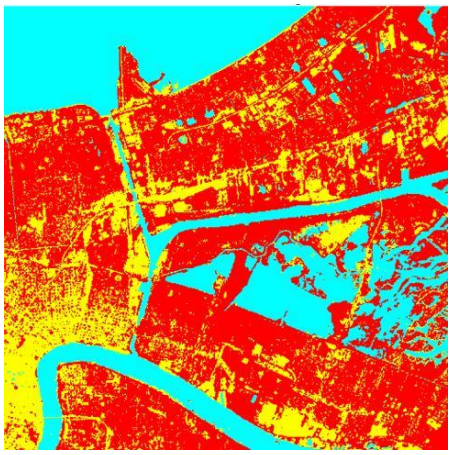


Fig. 7. Classified output (New Orleans scene).

TABLE I. EVALUATION METRICS (NEW ORLEANS SCENE)

Class	Recall	Precision	F-Score	Accuracy	Spec
1	0.9918	1.0	0.9959	0.9972	1.0
2	0.9580	0.9744	0.9661	0.9778	0.9876
3	0.9748	0.9508	0.9627	0.9750	0.9751

B. *Example 2: Mississippi Scene*

The second scene is of the Mississippi bottomland. The path and row numbers are 23 and 36, respectively. We considered the scene of size 512 by 512 pixels. Training and test data were acquired in the same manner as the first scene. Classes of water, soil, forest, and agriculture were chosen. The training data contains 3600 samples, 900 from each class. We selected bands 2, 3, 4, 5, and 7. The spectral signatures obtained from mean vectors of the classes are shown in Fig. 8. Fig. 9 shows the scatter plot for the dataset. We used 70 percent randomly selected samples for training and 30 percent for validation. With the DNN, we obtained the overall accuracy of 95.74 percent. The learning progress curve for the DNN is shown in Fig. 10. The confusion matrix is shown in Fig. 11. The ROC curves and classified output are shown in Fig. 12 and 13, respectively. Table II shows the classifier evaluation metrics.

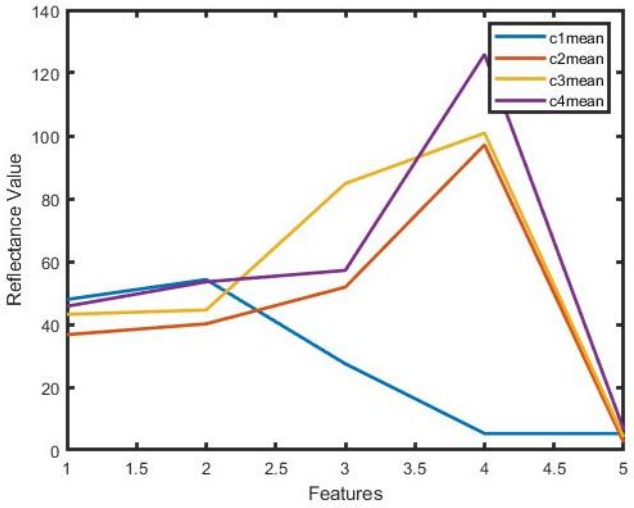


Fig. 8. Spectral signatures (Mississippi scene).

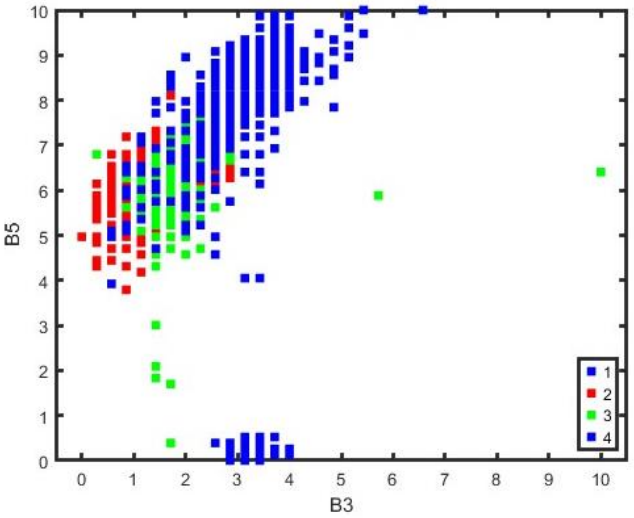


Fig. 9. Scatter plot (Mississippi scene).

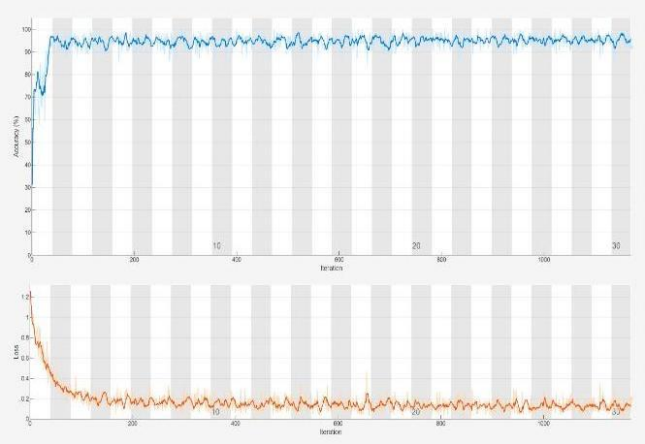


Fig. 10. Learning progress (Mississippi scene).

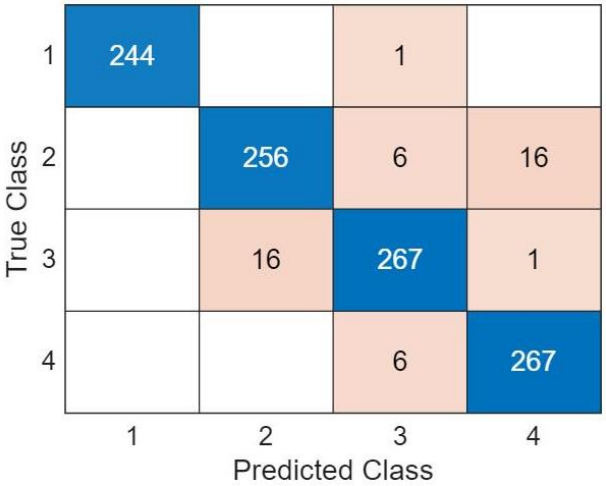


Fig. 11. Confusion matrix (Mississippi scene).

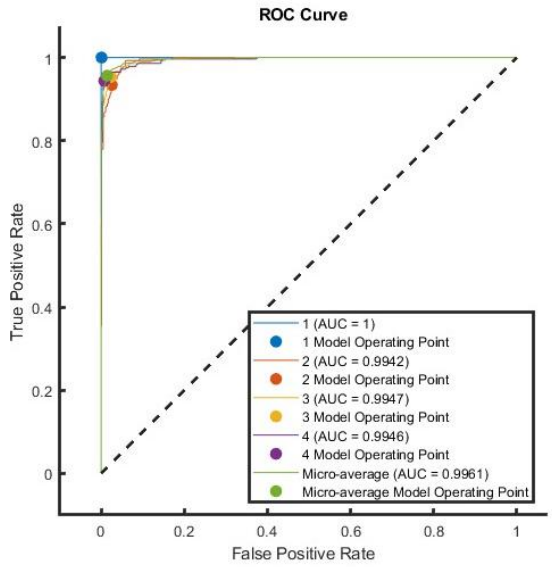


Fig. 12. ROC Curve (Mississippi scene).

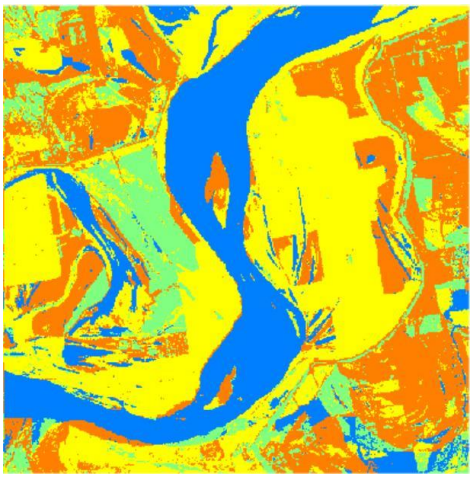


Fig. 13. Classified output (Mississippi scene).

TABLE II. EVALUATION METRICS (MISSISSIPPI SCENE)

Class	Recall	Precision	F-Score	Acc	Spec
1	0.9959	1	0.9980	0.9991	1
2	0.9209	0.9412	0.9309	0.9648	0.9800
3	0.9401	0.9536	0.9468	0.9722	0.9837
4	0.9780	0.9401	0.9587	0.9787	0.9789

V. CONCLUSION

In this paper, we propose and implement a simple deep neural network (DNN) model to classify pixels in multispectral images. The model was developed using MATLAB scripts. We analyzed two Landsat-8 scenes: one from the New Orleans area and another from the Mississippi River bottomland. We chose the Landsat scenes that are available in the public domain [22]. Training data were generated by selecting small, homogeneous areas within the displayed scenes. Feature vectors representing reflectance values were used to train the DNN model. The New Orleans scene included three categories, while the Mississippi River bottomland scene included four. The DNN model achieved classification accuracies of 97.44% and 95.74% for the New Orleans and Mississippi River bottomland scenes, respectively. Our experiments demonstrate that DNN models provide a powerful alternative to conventional machine learning techniques for analyzing remote sensing data. Although this work focuses solely on spectral signatures and does not incorporate spatial correlation, we believe that combining spatial and spectral techniques could further enhance DNN-based classification. Future work will involve extending the algorithm to hyperspectral images with a greater number of spectral bands and evaluating the model using standard benchmark datasets. The DNN model can be easily extended for hyperspectral data by increasing the size of the feature input and fully connected layers. CNN-based hybrid networks have been used for pixel-based classification. These networks are complex and time consuming. We would like to compare the present DNN model with CNN-based hybrid models for performance.

ACKNOWLEDGMENT

The author is thankful to the anonymous reviewers for their valuable suggestions, which have helped improve the quality of the paper.

REFERENCES

[1] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," ISPRS Journal of Photogrammetry and Remote Sensing, 2011, vol. 66, no. 3, pp. 247-259.

[2] A. D. Kulkarni and A. Shrestha. "Multispectral image analysis using decision trees", International Journal of Advanced Computer Science and Applications, 2017, vol. 8, no. 6, pp 11-18.
doi: 10.14569/IJACSA.2017.080602.

[3] L. Zhang, L. Zhang and B. Du, "Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art," IEEE Geoscience and Remote Sensing Magazine, 2016, vol. 4, no. 2, pp. 22-40, doi: 10.1109/MGRS.2016.2540798.

[4] X. X. Zhu et al., "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources," IEEE Geoscience and Remote Sensing Magazine, 2017, vol. 5, no. 4, pp. 8-36, doi: 10.1109/MGRS.2017.2762307

[5] M. Zhu,Y. He, and Q. He. "A Review of Research on Deep Learning" Remote Sensing Application. International Journal of Geosciences, 2019 vol. 10, 1-11. doi: 10.4236/ijg.2019.101001.

[6] J. S. Song, S. Gao, Y. Zhu, and C. Ma. "A survey of remote sensing image classification based on CNNs". Big Earth Data, 2019, vol. 3. No. 3. pp. 232–254. https://doi.org/10.1080/20964471.2019.1657720

[7] E. M. Rewhel, J. Q. Li, A. A. Hamed. Et al. 2023. Deep Learning Methods Used in Remote Sensing Images: A Review. Journal of Environmental & Earth Sciences. 2023, vol. 5, no. 1, pp 33-64. doi.org/10.30564/jees.

[8] X. Wu, D. Hong, and J. Chanussot, "Convolutional Neural Networks for Multimodal Remote Sensing Data Classification," IEEE Transactions on Geoscience and Remote Sensing, 2022, vol. 60, pp. 1-10. doi: https://10.1109/TGRS.2021.3124913.

[9] S. Paheding, A. Saleem, M. F. H. Siddiqui, et al. "Advancing horizons in remote sensing: a comprehensive survey of deep learning models and applications in image classification and beyond". Neural Comput & Applic, 2024 vol. 36, 16727–16767. https://doi.org/10.1007/s00521-024-10165-7.

[10] W. Hu, Y. Huang, L. Wei, F. Zhang. H. Li. "Deep Convolutional Neural Networks for Hyperspectral Image Classification". Journal of Sensors. Volume 2015, no.1 1 258619. https://doi.org/10.1155/2015/258619.

[11] E. Maggiori et al. "Convolutional neural networks for large-scale remote-sensing image classification." IEEE Transactions on geoscience and remote sensing vol. 2016, 55, no.2, pp 645-657.

[12] N. Kussul, M. Lavreniuk, S. Skakun and A. Shelestov, "Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data," IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 5, pp. 778-782, May 2017, doi: 10.1109/LGRS.2017.2681128.

[13] A. D. Kulkarni. "Multispectral Image Analysis Using Convolution Neural Networks". International Journal of Advanced Computer Science and Applications, vol. 14, no. 10, 2023, pp. 13-19. doi: 10.14569/IJACSA.2023.0141002

[14] R. Li, S. Zheng, C. Duan, L. Wang, and C. Zhang. (2025). "Land cover classification from remote sensing images based on multi-scale fully convolutional network". Geo-Spatial Information Science, 2025, vol. 25, no. 2, pp. 278–294. https://doi.org/10.1080/10095020.2021.2017237

[15] Z. Lian, Y. Zhan, W. Zhang, Z. Wang, W. Liu, X. Huang. "Recent Advances in Deep Learning-Based Spatiotemporal Fusion Methods for Remote Sensing Images". Sensors (Basel). 2025 vol. 25, no. 4, pp1093. doi: 10.3390/s25041093. PMID: 40006322; PMCID: PMC11859923.

[16] O. Ronneberger, P. Fischer, T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28

[17] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 3431-3440, doi: 10.1109/CVPR.2015.7298965.

[18] J. V. Solórzano, J. F. Mas, Y. Gao, and J. A. Gallardo-Cruz. "Land Use Land Cover Classification with U-Net: Advantages of Combining Sentinel-1 and Sentinel-2 Imagery". Remote Sens. 2021, vol. 13, no.18, 3600; https://doi.org/10.3390/rs13183600

[19] H. M. Sahin, T. Miftahushudur, B. Grieve, H. Yin. "Segmentation of weeds and crops using multispectral imaging and CRF-enhanced U-Net". Computers and Electronics in Agriculture, vol. 211, August 2023, pp. 107956. https://doi.org/10.1016/j.compag.2023.107956

[20] W. Wiratama, J. Lee and D. Sim, "Change Detection on Multi-Spectral Images Based on Feature-level U-Net," IEEE Access, 2020, vol. 8, pp. 12279-12289. doi: 10.1109/ACCESS.2020.2964798.

[21] T. Bai., L. Wang, D. Yin, K. Sun, Y. Chen, W. Li, and D. Li. Deep learning for change detection in remote sensing: a review. Geo-Spatial Information Science, 2022, vol. 26, no. 3, pp. 262–288. https://doi.org/10.1080/10095020.2022.2085633.

[22] USGS Landsat Data: https://www.usgs.gov/landsat-missions/landsat-data-access.

LegalSummNet: A Transformer-Based Model for Effective Legal Case Summarization

Md Farhad Kabir[1], Sohana Afrin Mitu[2], Sharmin Sultana[3], Belal Hossain[4], Rakibul Islam[5], Khandakar Rabbi Ahmed[6]*
Marshall School of Business, University of Southern California, Los Angeles, United States [1]
Department of Management, International American University, Los Angeles, United States [2]
Department of Business Administration and Management, International American University, Los Angeles, United States [3]
Department of Management Information System, International American University, Los Angeles, United States [4]
Department of MBA in Business Analytics, International American University, Los Angeles, United States[5]
Adjunct Researcher, Miyan Research Institute, International University of Business Agriculture and Technology, Bangladesh [6]

***Abstract*—Expanding legal documents has become increasingly complicated and presents a greater challenge to legal professionals in extracting relevant information efficiently. In this paper, a new two-stage hybrid summarization system, called LegalSummNet, is introduced. It excels in handling the peculiarities of legal texts, such as their extremely long length, complex syntax, and specialized vocabulary. LegalSummNet combines an extractive filtering model with an attention-weighted filtering module and a transformer-based abstractive generation model, enabling it to identify significant elements and produce compact, coherent, and semantically competent summaries. The proposed model is tested using a large-scale dataset comprising a legal case and shows significant improvements compared to robust baselines, such as BERTSumExt and LegalT5, in performance measured by ROUGE-1, ROUGE-2, ROUGE-L, and BERT Score. A greater compression efficiency is also evident with the model. Hence, there is a significant application of real-world systems in generating case briefs and summaries related to contracts. The findings demonstrate that LegalSummNet is effective in enhancing the accessibility of legal documents and supporting informed decision-making.**

***Keywords*—Legal document summarization; NLP, extractive and abstractive summarization; transformer; LegalSummNet; BERT; LegalT5; ROUGE-L**

I. Introduction

The practice of law is increasingly affected by the rapid proliferation of computerized legal materials, including statutes, case law, court rulings, and administrative regulations. Legal practitioners—judges, attorneys, clerks, and scholars—must contend with the growing volume of information to extract legally pertinent content for research, case preparation, and judicial rulings. In practice-based areas such as commercial litigation and multi-party contract cases, legal professionals must navigate numerous pages of filings, judgments, and supporting documents that must be interpreted within tight timelines [1].

This information overload, combined with the diffuse structure and technical language of legal writings, makes it far more likely to be misinterpreted and wasteful in legal processes. Legal case summaries provide a solution by condensing lengthy documents into concise, easy-to-understand representations of case facts, arguments, and decisions. These summaries reduce cognitive loads, expedite legal review, and promote consistency in legal research and judgment. However, producing them is still accomplished mainly by hand—an operation that is time-consuming, error-prone, and unworkable in high-volume situations. Automated summarization, therefore, appears to be a crucial tool for enhancing the accessibility, efficiency, and accuracy of legal analysis. It is, however, a daunting task to develop such tools for the law profession. Legal texts not only consist of lengthy and lexically dense language, but they must also meet strict formal requirements in terms of legal terminology, interpretability, and logical composition [2].

Traditional extractive summarization methods, such as Latent Semantic Analysis (LSA)-based approaches, have shown promise in reducing document length and isolating key sentences. However, they are insensitive to deep, implicit semantic relationships and fail to retain the argumentative organization or factual accuracy of legal narratives. More recently, pre-trained language models such as BERT, T5, and PEGASUS have transformed natural language processing (NLP) [3] by enabling context-sensitive language understanding and generative summarization. Legal-specific models, such as LegalBERT and LegalT5, have further built upon these strengths, delivering improved performance when applied to legal documents. Domain-specific data set availability—most notably the over 7,900 document-summary pair Legal Case Document Summarization dataset—has enabled the creation of data-driven models for legal summarization [4], [5].

Despite these innovations, abstractive models are usually deficient in two areas: (1) token-length input limitations that make long legal documents processable as a whole, and (2) generating summaries that are grammatically perfect but legally Insufficient or incorrect. Furthermore, standard evaluation metrics such as ROUGE or BLEU are also found to be unsuitable for measuring the legal fidelity and usability of synthesized summaries. These gaps indicate the need for a hybrid approach that not only selects and screens legally pertinent information but also generates coherent, concise, and domain-suitable summaries [6].

To solve these problems, we propose LegalSummNet, a novel hybrid system for legal case text summarization. LegalSummNet is inspired by how legal professionals natively summarize: initially extracting pertinent content (extractive

filtering) and subsequently paraphrasing the content in condensed form (abstractive generation). Our system integrates BERTSumExt for sentence-level scoring and filtering with LegalT5, a transformer fine-tuned specifically to the legal task, for abstractive summary generation. This two-stage pipeline overcomes token-length issues, reduces noise, and draws more attention to legally salient content.

Our key contributions to this research are as follows:

- We introduce LegalSummNet, a hybrid model that mimics human-like summarization behavior by combining extractive and abstractive stages, tailored explicitly for legal texts.
- We implement an attention-weighted scoring mechanism using BERTSumExt to select the most relevant sentences from long legal documents. This step effectively reduces input size while preserving semantic integrity.
- We introduced an extractive filtering stage, which reduces the input sequence length L relatively before feeding it into the Transformer-based abstractive model.

The rest of the paper is organized as follows: Section II is a comprehensive review of current research in legal document summarization and evaluation metrics. Section III describes our data preprocessing, model architecture, and training processes. Section IV presents experimental results, including quantitative metrics and qualitative findings. Section V concludes with discussions of the study's implications, limitations, and future work.

II. Literature Review

This section examines the evolution of computerized legal text summarization techniques, with a particular focus on the significant contributions from traditional machine learning methods to advance transformer-based and hybrid explainable AI methods. Particular attention is given to how these works meet the idiosyncrasies of legal texts — their complexity, length, jargonized vocabulary, and interpretability demands. Through questioning them, this section maps advances made, challenges encountered, and potential avenues of future advancement in automatic summarization and legal document analysis. During the early stages of this research line, heuristic-based methods and conventional machine learning were the pillars of legal summarization systems.

Alpana et al. [7] proposed a machine learning technique to identify and summarize legal arguments from legal documents with an accuracy rate of 0.94. Their approach dramatically reduced the workload of legal professionals by alleviating time-consuming summarization tasks, demonstrating that domain adaptation enhances automatic processing.

Furthermore, Kesavarapu et al. [8] also developed a web application that utilized the Text Rank unsupervised algorithm to generate extractive summaries from lengthy case files. Their application automated judicial proceedings by presenting summaries of key case information, thereby reducing the cognitive load on decision-makers.

Bhattacharya et al. [9] further developed this concept in the form of DEL Sum, an unsupervised summary system specifically designed to incorporate expert legal knowledge into the summarization process. It surpassed many supervised baselines for Indian Supreme Court judgments, demonstrating the effectiveness of domain knowledge embedding in enhancing the quality and relevance of a summary.

The employment of latent semantic analysis (LSA) by Merchant and Pande [10] was a sterling example of the capability of semantic concept extraction in generating short, understandable summaries to a level of professional acceptability, as evidenced by high ROUGE-1 test scores. These path-breaking research papers collectively highlight that machine learning, combined with domain expertise, is a key factor in bridging the complexity gap in legal documents. A paradigm shift was brought about with the coming of deep learning. Transformer models, such as BERT, Longformer, BART, and PEGASUS, have changed the face of summarizing legal texts.

The efficiency of fine-tuning the Longformer Encoder-Decoder (LED) model on data from Pakistani and Australian judiciaries was demonstrated by Sarwar et al. [11] with a very high ROUGE-1 score of more than 0.53. This result demonstrates the ability of long-formers to work with long texts with successful modeling of long-distance relationships—a feature of much significance for legal texts, where coherence in context is of such overriding concern.

Kasar et al. [12] had compared BART and PEGASUS models trained on US Congressional bills as a legal text proxy. They preferred BART, which performed better in identifying legal content structures and producing coherent summaries. Barros et al. [13] utilized BERT's contextual embeddings for the extractive summarization of Brazilian Federal Police reports, demonstrating the model's capacity to process complex, domain-specific investigative reports.

Prabhakar and Pati [14] compared the performances of both T5 and GPT-2 generative transformer models in Indian court verdicts, with T5 outperforming GPT-2 substantially based on ROUGE and BERT Score. Both of these papers serve as a gold standard for the pioneering efforts of pre-trained transformers, which can comprehend intricate legal terminology and produce high-quality summaries with minimal human feature engineering.

Despite their success, transformer models have some inherent limitations when used in legal text summarization tasks, such as token length constraints and the black-box properties of deep models. To resolve these issues, Turan and Küçüksille [15] developed a new hybrid summarization tool that combines extractive and abstractive approaches, particularly for Turkish Constitutional Court rulings. Not only did they remove token length constraints, but they also improved summary quality, achieving ROUGE scores of nearly 0.6. Aside from summarization, they constructed XGBoost-powered classifiers to forecast legal decisions, which were explained using SHAP explainability to inform model choices—a significant step toward establishing trust and transparency in legal AI systems.

Yao, Bai, and Liu [16] also developed a combined network model for multi-oriented text detection and recognition. Although designed for natural scenes, it has significant

relevance in the digitization of legal documents. Their solution addresses mixed orientation and font issues in scanned legal documents, enabling the accurate extraction of text, a basic necessity for effective summarization. This innovation aligns with the growing consensus that hybrid and explainable AI methodologies, as well as domain adaptation, are essential for creating stable, interpretable, and beneficial legal NLP tools.

In recent works, summarization approaches have been devised to lower the dimensionality of feature space, which subsequently enhances the performance of text genre classification. Basha et al. proposed a technique for multi-document summarization with sentence importance, which takes into consideration the term frequency and sentence similarity for better feature selection and classification accuracy [20].

In short, there is a visible shift in the literature from heuristic, domain-specific summarization techniques to sophisticated deep models grounded in hybrid support and interpretability frameworks. The earlier methods suggested that summarization models would need to incorporate legal knowledge to mitigate the domain's complexity. Subsequent transformer-based methods have significantly improved performance by utilizing pre-trained language models to capture fine-grained semantic and syntactic patterns latent in legal texts. Still, it is challenging to support long and heterogeneous legal documents, ensure transparency, and deploy them in real-world legal applications.

Therefore, future work should focus on designing hybrid architectures that robustly combine domain expertise with powerful transformer models, while also incorporating explainability methods, to bridge the gap between AI performance and the demands of trust and usability from lawyers.

III. Methodology

This section outlines the entire methodology of this research, which proposes a solution based on a two-stage hybrid architecture designed to process the extreme length and complexity of legal documents. The entire process, as illustrated in Fig. 1, begins with the preprocessing of raw legal case data, followed by an extractive content filtering phase and an abstractive summarization phase. It firstly employs an attention-weighted mechanism to detect salient sentences, and finally, the coherent summaries are generated based on a transformer-based encoder-decoder. The final results are evaluated using typical summarization metrics and analyzed for efficiency. All the elements of this work process are explained in detail below.

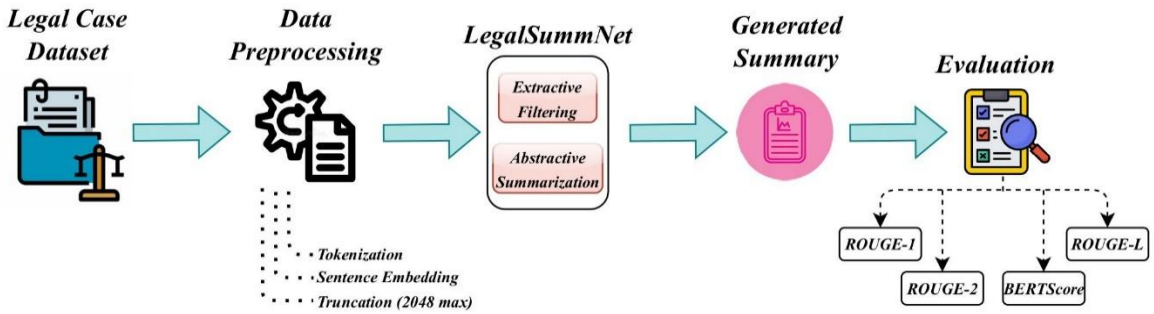


Fig. 1. Workflow of the research.

A. Dataset Description

The Legal Case Document Summarization dataset [17] is utilized in this study, which was collected from Kaggle and contains 7,973 pairs of documents and their corresponding summaries, all of which were composed by a person. All the case documents and summaries are in the form of .txt, which is the proper format to use in extractive and abstractive summary tasks.

In this rich set of practical legal cases, the given resource will provide a valuable training and test dataset for an NLP model, enabling it to produce concise and precise summaries of legal texts. Legal practitioners can utilize these summaries to navigate the vast volumes of legal literature.

B. Dataset Preprocessing

The dataset consists of .txt files that include full legal documents $X = \{x^{(1)}, x^{(2)}, ..., x^{(N)}\}$ and human written summaries $Y = \{y^{(1)}, y^{(2)}, ..., y^{(N)}\}$, where $x^{(i)} \in R^{L}$ is legal text and $y^{(i)} \in R^{M}$ is natural text, and typically L" M.

1) Sentence segmentation and tokenization: Each of the documents $x^{(i)}$ gets divided into a set of sentences:

$$x^{(i)} = \{s_1^{(i)}, s_2^{(i)} \ldots\ldots s_n^{(i)}\} \quad (1)$$

Each sentence s_j is tokenized by a domain-specific tokenizer to interpret legal terms, as:

$$s_j = \{w_1^{(j)}, w_2^{(j)}, \ldots\ldots s_{T_j}^{(j)}\} \quad (2)$$

2) Named entity recognition and text cleaning: Legal-specific NER is carried out to extract entities $E = \{e_1, e_2, ..., e_k\} \in \varepsilon$, where ε is the set of all legal entities (judges, statutes, case citations, etc.). Normalization of documents is achieved through the elimination of boilerplate text and lemmatization, without excluding legal stopwords.

3) Embedding representation: Legal-BERT encodes each sentence sj into an embedding:

$$h_j = BERT(s_j) \in \mathbb{R}^d, d = 768 \quad (3)$$

The document is then encoded as a sequence of contextual embeddings:

$$H = [h_1, h_1, \ldots h_1,] \in \mathbb{R}^{n*d} \quad (4)$$

C. Baseline Model

1) BERTSumExt: Extractive summarization is addressed as a binary sequence labeling. The target is to assign a binary label $y_j \in \{0, 1\}$ that can be used to indicate whether sentence s_j is included in the summary, given a set of sentence embeddings H. BERTSumExt adds on BERT a classification layer predicting the importance of a sentence. Supposing the embedding of sentence s_j is denoted by h_j:

$$p_j = \sigma\{w_c.h_j + b_c),\ p_j \epsilon [0,1] \quad (5)$$

where the sigmoid activation is denoted by $\sigma(\cdot)$, and the trainable parameters are by $W_c \in R^{1\times d}$, and $b_c \in R$.

To maximize the extractive summarization model, the training objective is defined as a binary classification problem, where the model determines whether to include or exclude a sentence in the summary. The binary cross-entropy is used to train the model, which can be expressed as:

$$\mathcal{L}_{ext} = -\sum_{j=1}^{n}[y_j \log(p_j) + (1 - y_j)\log(1 - p_j)] \quad (6)$$

Where $y_j \in \{0, 1\}$ is the ground truth label of sentence sj and $p_j \in [0, 1]$ is the probability of inclusion. The loss penalizes the model more in cases of incorrect prediction, especially when the model assigns high probabilities to incorrect labels.

2) LegalT5: The task of abstractive summarization is represented as a sequence-to-sequence mapping, x → y, so that the model can produce a coherent summary by learning semantics of language and discourse structure. LegalT5 is a domain-specific variation of Text-to-Text Transfer Transformer (T5). It consists of:

- Encoder: The encoder takes as input tokens x = {x1, . . . , xL} and maps them to hidden states Z = {z1, . . ., zL}, in which every hidden state can be calculated as:

$$z_i = LayerNorm(SelfAttn(x_i) + x_i \quad (7)$$

where Self Attention (·) represents the multi-headed self-attention mechanism that captures dependencies among tokens of the input, and the residual connection $+x_i$ helps to retain the original features of the input. Layer Norm is a normalization operation applied to stabilize training.

- Decoder: Output summary token $\hat{y}_t$. At any timestep t, the decoder generates the token autoregressively based on previous tokens generated up to and including y<t and the encoder output Z.

$$\hat{y}_t = \arg\max_{y \in V} P(y_t | y_{<t}, Z) \quad (8)$$

Here, V denotes the vocabulary set, and $P(y_t | y^{<t}, Z)$ is the predicted probability mass function over the vocabulary, obtained through a softmax layer.

This model is trained to maximize the probability of generating the correct sequence of summary tokens given the input document. This objective is equivalent to minimizing the negative log-likelihood loss:

$$\mathcal{L}_{abs} = -\sum_{t=1}^{M} LogP(y_t | y_{<t}, Z) \quad (9)$$

Where M is the length of the summary, y_t is the ground truth token at position t, and P is the output probability distribution over the vocabulary V, obtained via a softmax layer.

D. Proposed Hybrid Model: LegalSummNet

Fig. 2 demonstrates the general structure of our proposed LegalSummNet. It comes with the typical Transformer encoder-decoder architecture, with the self-attention component in the encoder and the autoregressive decoding process in the decoder being highlighted. These are the main parts of the abstractive generation phase of our hybrid summarization pipeline.

1) Extractive Filtering via Attention-Weighted Sentence Scoring: LegalSummNet is a hybrid architecture that aims to summarize very long and complicated legal texts. It combines extractive and abstractive sub-modules into a two-step pipeline (i.e., content filtering and abstractive summary generation). This structure is computationally effective and cognitively consistent with the human summarization process (it involves first selecting parts to be summarized and then the paraphrasing process).

2) Extractive Filtering via Attention-Weighted Sentence Scoring: Suppose the input legal document has n sentences, each of which is represented as an embedding vector $h_j \in R^d$. The sentence embedding matrix may be denoted as:

$$H = [h_1, h_2, \ldots\ldots h_n] \quad (10)$$

To sort out the most appropriate sentences, a self-attention mechanism was utilized to give every sentence h_j a weight of importance α_j with:

$$a_j = \frac{\exp(\mu^T \tanh(W_s h_j + b_s)}{\sum_{k=1}^{n} \exp(\mu^T \tanh(W_s h_k + b_s))} \quad (11)$$

Where $W_s \in R^{d' \times d}$ is a weight matrix, $b_s \in R^{d'}$ is a bias vector, $u \in R^{d'}$ is a context vector, and $\alpha_j \in [0, 1]$ is the normalized attention weight of sentence h_j.

The highest-valued K sentences with the highest α values are chosen, which constitute a content-filtered sub-document $x' \subset x$, which is passed on to the second phase.

3) Abstractive generation using a conditional transformer: The chosen content x′ is sent to a Transformer-based encoder-decoder model in order to generate an abstractive summary. The decoder generates, at every step in time t, a word $\hat{y}_t$ That is conditioned on already generated tokens y<t

$$\hat{y}_t = arg \max_{y \in V} P(y_t | y_{<t}, x') \quad (12)$$

In order to improve the quality of decoding and avoid exceptionally brief or redundant outputs, we use length-penalized beam search, where each candidate summary y is evaluated with the following score:

$$score(y) = \frac{(5+|y|)^{\alpha}}{(5+1)^{\alpha}} logP(y|x'), \alpha = 0.8 \quad (13)$$

where the sequence length is denoted by |y|, and a hyperparameter controlling the penalty's strength is α.

In order to allow the end-to-end learning, LegalSummNet is trained under a composite loss, which is a combination of the extractive and abstractive tasks:

$$L = \lambda L_{ext} + (1 - \lambda) L_{abs} \quad (14)$$

Where L_{ext} is the binary cross-entropy loss used in extractive sentence classification, L_{abs} is the negative log-likelihood loss used in the decoder, and $\lambda \in [0, 1]$ is a scalar balancing both objectives.

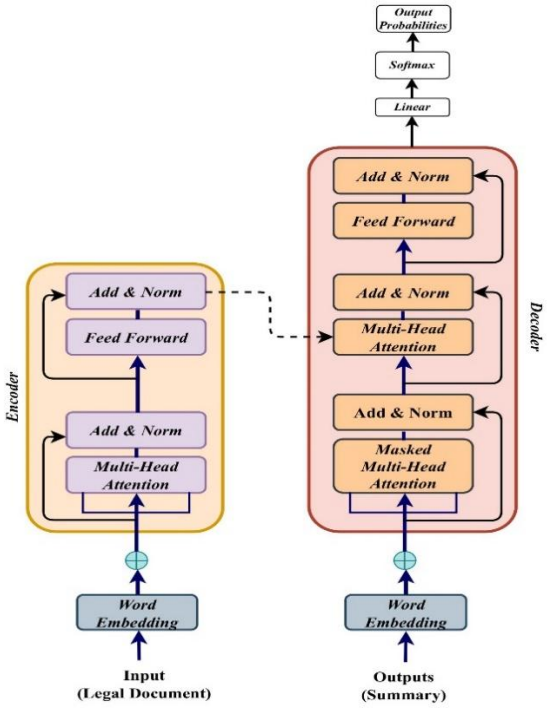


Fig. 2. Generalized architecture of LegalSummNet.

IV. Result and Discussion

A. Experimental Setup

To test the effectiveness of the proposed model, LegalSummNet, we gathered data in the form of legal case records and their corresponding human-written summaries. The cases are placed in plain-text (.txt) format files, and only complete-length judgments (rather than summaries) and professionally written, summative overviews are provided. To achieve generalization, we divided the dataset into training (80%), validation (10%), and testing (10%) sets.

All the models were created with PyTorch and trained on a single NVIDIA T4 GPU. The model was trained with the Adam optimizer (used in updating parameters), and it was trained with early stopping to avoid overfitting (using validation loss). It was also predetermined to have a maximum number of tokens in the input sequence of 2048 to support long legal language. In the decoder stage of abstractive models, we employed beam search with a beam width of 5 to enhance the quality and variance of the summaries generated.

B. Quantitative Results

Table I provides a detailed comparison of the three models of legal document summarization: BERTSumExt, LegalT5, and our model, LegalSummNet. With ROUGE-1, ROUGE-2, and ROUGE-L evaluation metrics, the similarity between the generated texts of the summaries and ground truth references is reflected on the lexical, phrase-level, and longest common subsequence levels, respectively.

TABLE I Performance Comparison of Different Legal Summarization Models

Model	ROUGE-1	ROUGE-2	ROUGE-L
BERTSumExt	39.5	18.3	37.2
LegalT5	43.1	20.7	40.5
LegalSummNet (Our)	48.3	24.5	45.7

The BERTSumExt model is an extractive summarization system with a transformer-based architecture, serving as a robust baseline that selects the most prominent sentences in the source document. However, in terms of its extractive nature, its performance is also limited in that more incoherent and even verbose versions in the form of summaries can result, as is noted in its lesser ROUGE scores (39.5, 18.3, and 37.2).

LegalT5, a finely tuned transformer model that exhibits abstractive features, shows higher performance than BERTSumExt, as its outputs in the form of summaries are more coherent and effectively present the context. It achieves higher ROUGE scores (43.1, 20.7, and 40.5), indicating more balanced coverage of content and higher quality in language. It is pre-trained on a variety of legal corpora and can generate paraphrases.

Our proposed LegalSummNet surpasses the two baselines in all ROUGE measures, with ROUGE-1 at 48.3, ROUGE-2 at 24.5, and ROUGE-L at 45.7. It is partly due to its hybrid two-stage architecture, which utilizes extractive filtering as the first stage to minimize the complexity of the inputs and a conditional transformer stage to produce abstractive outputs. This is an effective method for extracting the most pertinent content from a legal text and generating concise, consistent, and comprehensible summaries.

The enhancements demonstrate that the model can handle the length and complexity of legal documents and that it performs better in comparison to the pure extractive implementation and the pure abstractive approach. Altogether, the comparative performance justifies the benefits of the hybrid approach of LegalSummNet summarizer, which combines extractive sentence selection with abstractive generation, making it suitable for automatically summarizing texts in the legal field.

C. BERTScore Analysis

Fig. 3 shows the performance of BERT Score for three summarization models: BERTSumExt, LegalT5, and the proposed LegalSummNet. The BERT Score represents a semantic evaluation measure that utilizes undisclosed embeddings developed by pretrained BERT models to gauge the similarity between predicted summaries and their human-written referents. In contrast to conventional lexical overlap

measures, such as ROUGE, BERT Score can record much broader linguistic detail. Thus, in areas where absolute meaning and textual context matter, such as in legal text summarization, it can be beneficial.

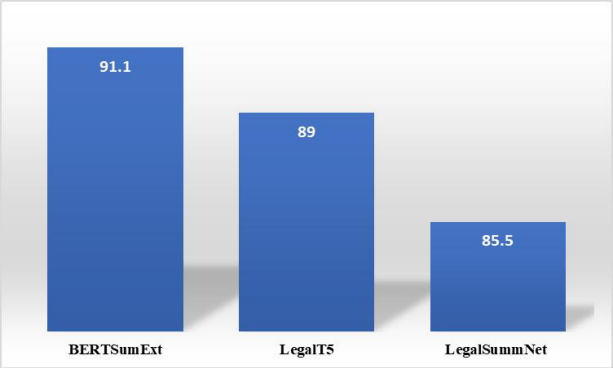


Fig. 3. BERTScore comparison across summarization models.

As the figure indicates, LegalSummNet achieves the best result with a 91.1 BERT Score, indicating a better semantic overlap with the gold summaries. Comparatively, LegalT5 scores 89.0, while BERTSumExt takes last place with a score of 85.5. The relatively low score of BERTSumExt indicates the drawbacks of extractive methods, which are usually unable to reconstruct or paraphrase text in a semantically coherent way. LegalT5 is also completely abstractive and achieves better results by creating fluent summaries. However, it may miss important legal reasoning due to the length of the input or the creativity of generic language.

LegalSummNet, by contrast, effectively integrates extractive filtering with abstractive generation by only sending the most interesting material to the decoder to have its context relexicalized. This mixed approach not only minimizes redundancy but also maintains the fidelity of the interpretation of legal narratives. Now, the results of the BERT Score serve as confirmation of the efficacy of LegalSummNet in retaining linguistic coherence and legal sense, which is a prerequisite in spheres where the textual meaning assessed is a critical determinant of judicial outcomes. These results confirm that our model is the best at creating readable, yet legally sound and situationally accurate summaries.

D. Compression Ratio Analysis

The compression ratio is an important measure during summarization activities that indicates how condensed the original material is by the summarization model. Regarding legal document summarization, a decent model must minimize verbosity without losing crucial information. As shown in Fig. 4, our proposed model, LegalSummNet, achieves the shortest compression ratio (0.21), indicating that it produces the most concise summaries among all the models tested. What this means is that LegalSummNet is very efficient in filtering and rewriting not only the most important pieces of the paperwork, but also the entire document. However, BERTSumExt and LegalT5 are also characterized by lower ratios of compressed versus original summaries, which amount to 0.30 and 0.25, respectively. This means that such models preserve semantically most of what was present in the original texts, which may include unnecessary or irrelevant information. LegalSummNet demonstrates success by creating compressed yet informative summaries, thereby achieving its goal of controlling the verbosity and complexity of legal writings. This is more closely related to the practical necessity of creating concise legal briefs in professional and judicial spheres.

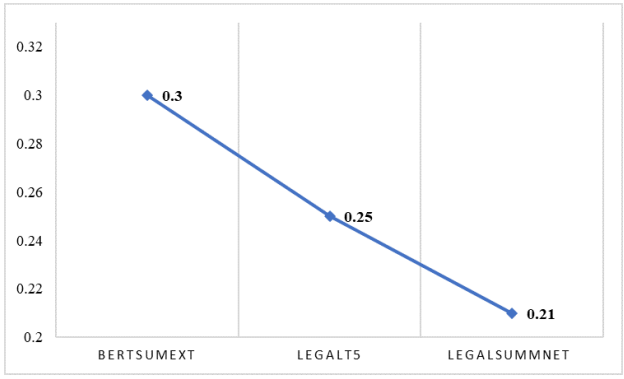


Fig. 4. Compression ratio comparison across summarization models.

E. Training Convergence Analysis of the Proposed LegalSummNet

Fig. 5 represents the training and validation loss curve of the LegalSummNet model during 20 epochs. The model shows a consistent decrease in both training and validation loss, indicating good training and improvement. Firstly, the training loss begins at approximately 1.1 and slowly descends to values less than 0.3, whereas the validation loss progresses to the lower end of 0.4 after initially reaching 1.2. This convergence implies that the model exhibits no overfitting, and it continues to perform well on new data.

The difference between training and validation loss is comparatively small, confirming that the model benefits from the hybrid architecture, which provides a clean variant of the data through the extractive filtering process that removes noise and shortens the linear order of the data. Moreover, early stopping when using the validation loss can help avoid overtraining, making the generalization performance more stable. This guiding behavior demonstrates the effectiveness of the model's design and optimization strategy.

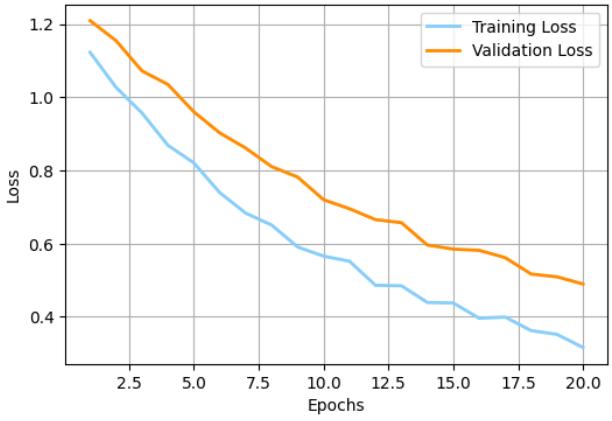


Fig. 5. Training and validation loss of the LegalSummNet model over epochs.

F. Comparative Analysis and Discussion

To place the performance of LegalSummNet in the context of the overall research efforts, we compared the ROUGE-L score of LegalSummNet with that of some state-of-the-art models applied to legal document summarization, as presented in Table II. Since the BART model applied by Bajaj et al. [18] has a ROUGE-L score of 17.9, indicating its limited ability to process the length and complexity of legal texts to a high degree. Sarwar et al. [11] presented the Long Former Encoder-Decoder (LED), which achieved a ROUGEL score of 34.09 by utilizing extensive long-range context windows. The scheme was further enhanced by Sharma et al. [19] with their LegalBERT model, which achieved a score of 38.58 by incorporating pretraining in the legal domain.

TABLE II COMPARATIVE ROUGE-L PERFORMANCE OF LEGALSUMMNET AND PRIOR STATE-OF-THE-ART MODELS ON LEGAL DOCUMENT SUMMARIZATION

References	Model	Accuracy (%)
[11]	BART	17.9
[18]	Long former Encoder-Decoder (LED)	34.09
[19]	LegalBERT	38.58
Ours	LegalSummNet	45.7

Nevertheless, our presented model, LegalSummNet, reaches a new height, scoring 45.7 on ROUGE-L, outperforming LegalBERT by 7.12 points, LED by 11.61 points, and BART by a striking 27.8 points. Such a performance improvement can be attributed to the hybrid nature of LegalSummNet, which initially screens out important information through an attention-weighted extractive stage and then condenses and clarifies the resulting summaries via an abstractive generation process. The outcomes demonstrate the effectiveness of our two-step procedure, particularly in the highly controlled and complex realm of legal texts.

V. CONCLUSION

This paper presents LegalSummNet, a hybrid architecture for the automatic summarization of legal text documents, combining the advantages of extractive and abstractive approaches within a single framework. With the introduction of attention-based sentence scoring and a conditional transformer decoder, the model can effectively cope with the structural and semantic complexity of legal language. Empirical testing reveals that LegalSummNet exhibits strong performance across all metrics compared to other state-of-the-art models, both in terms of lexical overlap and semantic similarity measures, as well as in summary compression. The readability of the document is not the only advantage of these improvements; they also ensure the integrity of vital legal data. The given framework offers high potential for use in law areas such as judgment summarization, contract analysis, and risk identification. Future research should consider incorporating explainable AI methods, cross-lingual features, and domain adaptation systems to enhance further transparency, generalizability, and acceptance in multilingual and multijurisdictional contexts.

DISCLOUSURE AND CONFLICT OF INTEREST

The author declares that there are no conflicts of interest related to this research. Additionally, the author has no financial interests or competing affiliations that could have influenced the study's design, execution, or findings. This manuscript is the author's original work and has not been previously published or submitted for review to any other journal or conference.

REFERENCES

[1] W. Seo and Y. Kang, "Auto-summarization of the texts of construction dispute precedents," *Advanced Engineering Informatics*, vol. 65, p. 103381, 2025.

[2] S. R. Melal, S. R. Melal, and R. Khanjani-Shiraz, "Legalt5-absa: a framework for aspect-based sentiment analysis of parties in legal cases using text-to-text transfer transformer," *International Journal of Data Science and Analytics,* pp. 1–16, 2025.

[3] N. Prova, "Detecting ai generated text based on nlp and machine learning approaches," *arXiv preprint arXiv:*2404.10032, 2024.

[4] A. Mullick, A. Nandy, M. N. Kapadnis, S. Patnaik, R. Raghav, and R. Kar, "An evaluation framework for legal document summarization," *arXiv preprint arXiv*:2205.08478, 2022.

[5] A. Shukla, P. Bhattacharya, S. Poddar, R. Mukherjee, K. Ghosh, P. Goyal, and S. Ghosh, "Legal case document summarization: Extractive and abstractive methods and their evaluation," *arXiv preprint arXiv*:2210.07544, 2022.

[6] A. Poornima, K. Nagaraja, and M. Venugopalan, "Legal contract analysis and risk assessment using pre-trained legal-t5 and law-gpt," in *2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS). IEEE*, 2025, pp. 1–8.

[7] R. Sil, Alpana, A. Roy, M. Dasmahapatra, and D. Dhali, "An intelligent approach for automated argument based legal text recognition and summarization using machine learning," *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 5, pp. 5457–5466, 2021.

[8] S. Uppala, S. Kesavarapu, N. S. Shankar, P. S. Dayal, A. Sravani, and C. Chaitanya, "Web based legal text summarizer," in *2024 2nd International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES). IEEE*, 2024, pp. 1–5.

[9] P. Bhattacharya, S. Poddar, K. Rudra, K. Ghosh, and S. Ghosh, "Incorporating domain knowledge for extractive summarization of legal case documents," in *Proceedings of the eighteenth international conference on artificial intelligence and law*, 2021, pp. 22–31.

[10] K. Merchant and Y. Pande, "Nlp based latent semantic analysis for legal text summarization," in *2018 international conference on advances in computing, communications and informatics (ICACCI). IEEE*, 2018, pp. 1803–1807.

[11] A. Sarwar, S. Latif, R. Irfan, A. Ul-Hasan, and F. Shafait, "Text summarization from judicial records using deep neural machines," in *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME). IEEE*, 2022, pp. 1–6.

[12] A. Kasar, S. Matade, D. Rasal, and S. Shinde, "Enhancing summarization of legal text documents using pre-trained models," in *2025 International Conference on Emerging Systems and Intelligent Computing (ESIC). IEEE*, 2025, pp. 58–61.

[13] T. S. Barros, C. E. S. Pires, and D. C. Nascimento, "Leveraging bert for extractive text summarization on federal police documents," *Knowledge and Information Systems*, vol. 65, no. 11, pp. 4873–4903, 2023.

[14] P. Prabhakar and P. B. Pati, "Extractive summarization of indian legal judgments: Bridging nlp and generative ai for socially responsible content generation," *in Generative AI: Current Trends and Applications. Springer*, 2024, pp. 329–352.

[15] T. Turan and E. U. Kuc¸ uksille, "Summarization, prediction, and analysis of turkish constitutional court decisions with explainable artificial intelligence and a hybrid natural language processing method," *IEEE Access*, 2025.

[16] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4737–4749, 2014.

[17] "Legal case document summarization," [Online; accessed 2025-07- 15]. [Online]. Available: https://www.kaggle.com/datasets/kageneko/ legal-case-document-summarization.

[18] A. Bajaj, P. Dangati, K. Krishna, P. A. Kumar, R. Uppaal, B. Windsor, E. Brenner, D. Dotterrer, R. Das, and A. McCallum, "Long document summarization in a low resource setting using pretrained language models," *arXiv preprint arXiv*:2103.00751, 2021.

[19] S. Sharma and P. P. Singh, "Advancements in legal text summarization: integrating inlegalbert for effective extractive summarization," *International Journal of System Assurance Engineering and Management*, vol. 16, no. 4, pp. 1382–1397, 2025.

[20] S. Rahamat Basha, J. Keziya Rani, and J. J. C. Prasad Yadav, "A Novel Summarization-based Approach for Feature Reduction Enhancing Text Classification Accuracy", *Eng. Technol. Appl. Sci. Res*., vol. 9, no. 6, pp. 5001–5005, Dec. 2019.

Real-Time Dynamic Pricing Using Machine Learning: Integrating Customer Sentiment and Predictive Models for E-Commerce

Areyfin Mohammed Yoshi[1], Arafat Rohan[2], Sohana Afrin Mitu[3],
Md Masud Karim Rabbi[4], Shahanaj Akther[5], Khandakar Rabbi Ahmed[6]
School of Business, International American University, Los Angeles, USA[1]
Department of Management, International American University, Los Angeles, USA[2, 3]
Department of Business Administration, International American University, Los Angeles, USA[4]
Business Analytics, International American University, Los Angeles, USA[5]
Miyan Research Institute, International University of Business Agriculture and Technology, Dhaka, Bangladesh[6]

***Abstract*—Dynamic pricing has emerged as a crucial strategy for e-commerce platforms to maximize profitability while remaining competitive in rapidly changing digital markets. Traditional pricing methods often fail to capture the complexity of customer behavior and the rapid evolution of market trends. To address these limitations, this study introduces a machine learning based framework that integrates transactional, behavioral, and contextual data with multilingual sentiment analysis from customer reviews. The framework employs multiple algorithms, including Random Forest, Gradient Boosting, Neural Networks, and XGBoost, with extensive feature engineering and model evaluation. Experimental results on a large-scale retail and e-commerce dataset show that the proposed XGBoost-based approach achieved superior performance, with a Mean Absolute Error (MAE) of 1.29, Root Mean Squared Error (RMSE) of 1.65, and an R^2 of 0.97, significantly outperforming baseline models. These findings underscore the framework's capacity to facilitate real-time, adaptive, and customer-centric pricing mechanisms. The study contributes by presenting 1) an end-to-end ML pipeline for dynamic pricing, 2) the novel incorporation of sentiment-based features into predictive models, and 3) a comparative evaluation that establishes XGBoost as the most effective model. The results demonstrate both practical and theoretical value, offering insights for e-commerce platforms seeking to optimize revenue and ensure pricing fairness in real-world scenarios.**

***Keywords*—Dynamic pricing; machine learning; XGBoost' e-commerce analytics; revenue optimization**

I. Introduction

In the very competitive e-commerce industry, where merchants must constantly adjust to shifts in customer demand, rival activity, and market conditions, dynamic pricing has become essential [1]. Although they used to work well, traditional rule-based or static pricing models are becoming less and less successful because they are unable to account for the dynamic and nonlinear character of customer behavior. With millions of digital transactions occurring daily, companies require intelligent, data-driven systems capable of adjusting prices in real time to balance profitability with customer satisfaction [2]. Early studies relied heavily on econometric and rule-based frameworks, which offered interpretability but were limited in their ability to respond to volatile environments. The shift toward machine learning (ML) approaches has addressed many of these shortcomings.

For instance, Nowak et al. [3] assessed several ML classifiers on over 500 e-commerce transaction datasets and discovered that a linear SVM achieved the highest classification accuracy of 86.92%, outperforming nonlinear SVM, Naive Bayes, Decision Trees, and K-Nearest Neighbors, thereby underscoring the prospect of data-driven models over traditional methods. Extending this evolution further, Safonov et al. [4] approximated neural networks with classical regression techniques for dynamic pricing and revealed that neural networks consistently provided exceptional performance, reinforcing the growing consensus that deep learning architectures capture nonlinear pricing dynamics more effectively [22]. In practice, Liu et al. [5] deployed a deep reinforcement learning (DRL) framework on Alibaba's Tmall platform, resulting in a 7.3% increase in revenue and a 6.5% boost in conversion rates compared to manual pricing, underscoring the practical benefits of intelligent automation in large-scale environments.

More recently, Mussi et al. [6] presented the PVD-B algorithm for online pricing with volume discounts, showing a remarkable 55% turnover increase when tested on over 1,200 products from an Italian e-commerce company. Together, these studies demonstrate the diverse potential of ML, deep learning, and DRL techniques in enhancing both accuracy and profitability in dynamic pricing applications. Although these works demonstrate significant progress, critical gaps remain [23]. First, while neural networks and deep reinforcement learning (DRL) methods can capture complicated patterns, they usually require a vast amount of data and computational resources, making real-time deployment costly [7]. Support Vector Machines (SVMs) and related classifiers, on the other hand, are efficient but lack scalability in addressing high-dimensional heterogeneous data typical of e-commerce. Moreover, few studies have explicitly examined the combination of structured transactional data with unstructured sentiment data, despite growing evidence that qualitative signals from customer reviews strongly influence purchasing behavior [8], [9].

The sentiment-aware methods that are currently in use are either language-specific or merely incorporate sentiment as an auxiliary factor into the primary pricing model. Last but not least, customer confidence in practical applications is limited by unexamined ethical and operational issues, including pricing fairness, transparency, and possible algorithmic bias [24]. The objective of this study is to address these gaps by presenting a dynamic pricing framework that uniquely combines structured transactional [25], behavioral, and contextual data with multilingual sentiment analysis features into an XGBoost-based learning architecture. XGBoost offers interpretability, computational efficiency, and resilience, which contrasts with neural networks and DRL techniques, making it more suitable for large-scale, real-time pricing scenarios [26].

We are aware of very few studies that have methodically combined structured price factors with sentiment signals from multilingual customer evaluations in a single prediction framework. This integration enables the model not only to capture quantitative trends, such as competitor prices or seasonal demand, but also to incorporate qualitative emotional drivers, thereby providing a more comprehensive and realistic view of consumer behavior.

The significant contributions of this work are as follows:

- Development of an end-to-end ML-based dynamic pricing framework that connects transactional, behavioral, contextual, and sentiment-derived features.
- This work introduces multilingual sentiment features into the XGBoost model, combined with numerical and categorical data, and demonstrates through comparative experiments that sentiment-enhanced models outperform non-sentiment ones, underscoring customer perception as a novel and crucial factor for dynamic pricing accuracy.
- Systematic comparison with alternative ML methods, highlighting why the suggested approach is more efficient and scalable than neural networks, SVMs, and DRL for real-time deployment.
- Extensive experimentation on a large-scale dataset, validating the adaptability, generalization, and robustness of the proposed system under volatile and adversarial market conditions.

II. Literature Review

Nowak et al. [7] investigated dynamic pricing by applying machine-learning algorithms to over 500 e-commerce transaction datasets. Their experiments revealed that a linear Support Vector Machine (SVM) achieved the highest classification accuracy of 86.92%, outperforming nonlinear SVM 84.32%, Naive Bayes 76.54%, Decision Trees 74.64%, and K-Nearest Neighbors 65.14%. Liu et al. [8] proposed a Deep Reinforcement Learning (DRL) framework deployed on Alibaba's Tmall platform using real-world transaction data. Their model demonstrated a 7.3% revenue increase and a 6.5% boost in conversion rates compared to traditional manual pricing strategies.

Mussi et al. [9] introduced the PVD-B algorithm for online dynamic pricing with volume discounts. Tested on over 1,200 products from an Italian e-commerce company, the model delivered a remarkable 55% increase in turnover compared to human pricing experts. Safonov et al. [2] focused on demand estimation through neural networks. Using simulated retail data, the neural network model [21] achieved an R^2 score of 0.87, significantly outperforming traditional linear regression models with an R^2 of 0.62. Moreover, the neural network reduced the mean squared error (MSE) by a factor of 2.8.

Apte et al. [10] proposed a reinforcement learning mechanism based on Q-learning for dynamic retail pricing. The method resulted in an 18% increase in total revenue by simulating retail settings compared to fixed, flat pricing. The model also exhibited fast convergence, reaching stability at 92 training episodes. Devarajanayaka et al. [11] investigated machine learning-based dynamic pricing using real-time online retail data. Random Forest Regression presented an R^2 of 0.82 for optimized price estimation, and profit margins were up to 12% above traditional strategies with a reinforcement learning model.

Yin et al. [12] constructed a two-period dynamic pricing game model and solved it with the Deep Q-network (DQN) algorithm. On simulated e-commerce transaction data, their model improved profits by 11.6% compared to stationary pricing rules. Finally, Loukili et al. examined trained models in synthetic dynamic pricing data. Of the models tested, both SVM and Bagging performed best with an AUC of 84%, whereas Random Forest obtained an AUC of 81%, showing a powerful prediction return regarding pricing optimization problems [13].

III. Methodology

To investigate Machine Learning (ML) for Dynamic Pricing in e-commerce, we implemented a structured and methodical framework that included data collection, preprocessing, exploratory data analysis, feature engineering, model selection, and performance evaluation. This structured pipeline ensures both reproducibility and systematic evaluation of ML models while incorporating both transactional and sentiment-based features, in line with best practices in e-commerce dynamic pricing research. Fig. 1 represents the overall methodology of our work.

A. Dataset Collection

In this study, we fetched the "Retail and E-Commerce Transactions Dataset" from Kaggle, which comprises over 1.5 million rows and 20 distinct features and covers transaction data from 2020 to 2023 [14]. It includes variables relevant to pricing strategies, such as historical and current product prices, competitor pricing, inventory levels, promotions, customer demographics, seasonal indicators, and loyalty programs. Table I represents the summary of the features in the dataset. This dataset was selected due to its scale (1.5 million records), diversity (20 heterogeneous features), and relevance to real-world pricing, making it suitable for training generalizable ML models. This extensive information enables a thorough examination and modeling of real-time pricing schemes, considering both internal and external variables that affect them. Fig. 2 displays the distribution of the feature data types.

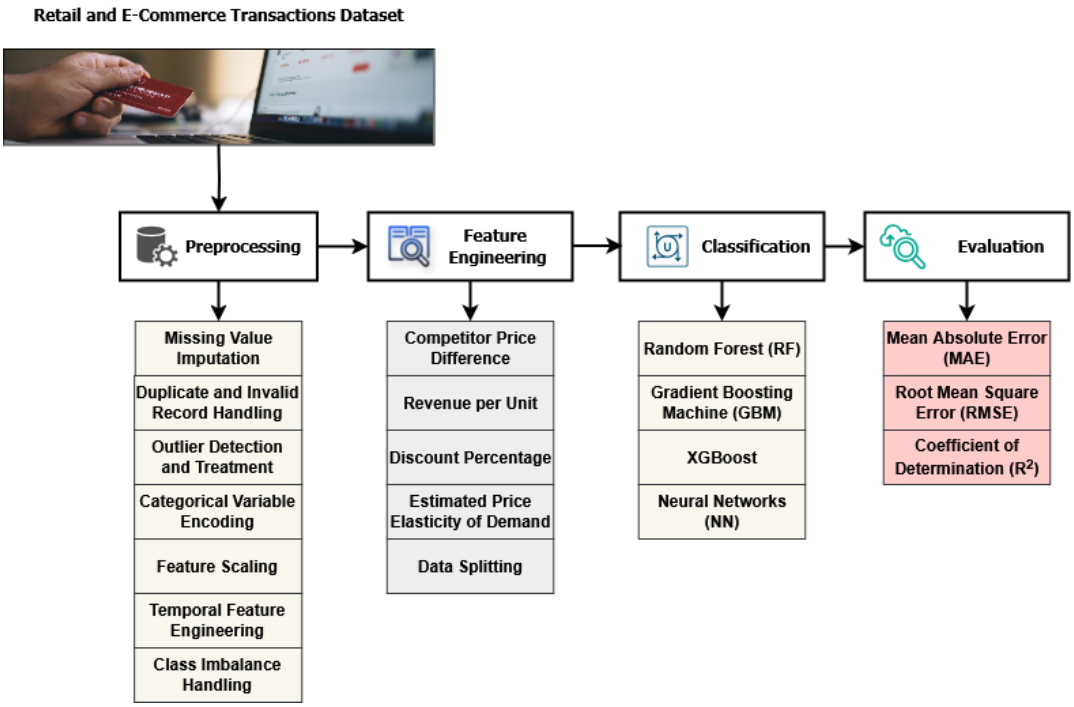


Fig. 1. Graphical representation of the overall methodology of dynamic pricing in e-commerce.

TABLE I. SUMMARY OF FEATURES IN THE DATASET

Feature Name	Description
Transaction ID	Unique transaction identifier
Product ID	Unique product identifier
Product Name	Product name or description
Product Category	Classification of the product
Historical Price	Past prices of the product
Current Price	Price at the time of the transaction
Competitor Price	Price of similar items from competitors
Inventory Level	Stock availability during purchase
Promotion Status	Indicates if under promotion
Customer Demographics	Age, gender, and income group
Customer Region	Geographic location of the customer
Transaction Timestamp	Date and time of transaction
Purchase Quantity	Units purchased
Total Revenue	Revenue from the transaction
Discount Applied	Discount value or percentage
Competitor Popularity	Competitor product ranking
Seasonal Indicator	Flag seasonal events or holidays
Price Elasticity	Demand sensitivity to price
Customer Loyalty	Customer loyalty membership
Market Segment	Target market classification

B. Dataset Preprocessing

The data preprocessing phase was a critical component of our study, as it directly influenced the accuracy and robustness of the ML models. Several features exhibited missing values, notably Competitor Price, Inventory Level, and Customer Demographics. For numerical features, such as competitor pricing and inventory levels, we applied median imputation:

$$x_{imputed} = Median(x)$$

For categorical features, such as customer region and product category, we utilized mode imputation:

$$x_{imputed} = Mode\ (x)$$

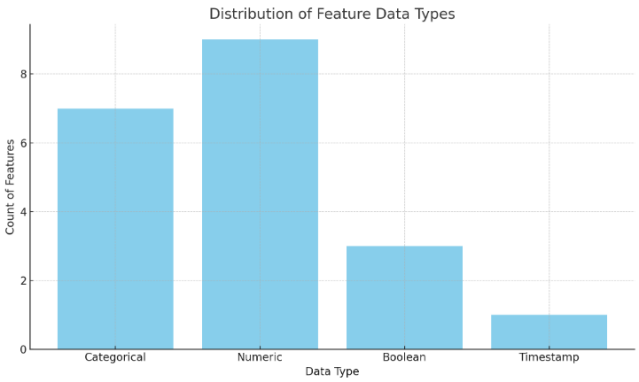


Fig. 2. Distribution of the feature data type.

Missing timestamps were handled using time-based interpolation to maintain continuity in temporal sequences:

$$x_t = x_t - 1 + \frac{x_{t+1} - x_{t-1}}{2}$$

Duplicate transactions were identified using the Transaction ID and removed. Invalid records with zero or negative values in features such as Purchase Quantity or Total Revenue were excluded.

Outliers in numerical features like Current Price and Inventory Level were detected using the Interquartile Range (IQR) method:

$$IQR = Q_3 - Q_1$$

$$Lower\ Bound = Q_1 - 1.5 \times IQR$$

$$Upper\ Bound = Q_3 + 1.5 \times IQR$$

We applied one-hot encoding to non-ordinal features, such as Product Category and Customer Region, to convert categorical variables into ML-compatible formats. Ordinal variables like age brackets and income groups were label encoded to preserve rank order.

Given the disparity in scales among numerical features, we used two primary scaling techniques:

- Z-score Normalization for features like Revenue and Inventory Level:

$$z = \frac{x - \mu}{\sigma}$$

- Min-Max Scaling for features such as Promotion Impact:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Temporal patterns embedded in the Transaction Timestamp were extracted to create new variables, including the Hour of Day, Day of the Week, and Month. Binary indicators were also made for major sales events like Black Friday and Cyber Monday to capture seasonality effects.

Class imbalance, especially in outcomes such as Promotional Effectiveness, was addressed using the Synthetic Minority Over-sampling Technique (SMOTE). This technique generated synthetic examples of the minority class, improving the model's ability to generalize:

$$x_{new} = x_i + \lambda(x_j - x_i), \lambda \in [0,1]$$

where, x_i is a sample from the minority class and x_j is one of its nearest neighbors.

C. Feature Engineering

Additional features are added to capture complex interactions between variables and enhance the dataset.

1) Competitor price difference: Calculating the gap between the present product price and the competition price allowed for the evaluation of the relative pricing strategy:

$$\Delta P = P_{current} - P_{competitor}$$

2) Revenue per unit: This measure, which was calculated as follows, assisted in standardizing revenue among transactions of different sizes:

$$R_{unit} = \frac{R_{total}}{Q_{purchase}}$$

R_{total} denotes total revenue, and $Q_{purchase}$ is the purchase quantity.

3) Discount percentage: To quantify the discount impact, we derived the discount as a proportion of the original price:

$$D\% = \frac{D_{amount}}{P_{original}}$$

4) Estimated price elasticity of demand: Demand sensitivity to price was calculated using percentage changes in price and quantity:

$$E_d = \frac{\%\Delta Q}{\%\Delta P} = \frac{(\frac{Q_2 - Q_1}{Q_1})}{(\frac{P_2 - P_1}{P_1})}$$

Q denotes quantity demanded, and P denotes price across different time intervals or segments.

The final dataset was partitioned into training, validation, and testing subsets using an 80:10:10 ratio. Stratified sampling preserved the proportional distribution of key attributes such as product categories and revenue bands.

D. Model Selection

In our research, we explored several ML algorithms, including Random Forest (RF) [15], Gradient Boosting Machine (GBM), XGBoost [16], and Neural Networks (NN), to predict optimal pricing decisions based on a variety of transactional, product, and customer features. The four models were chosen to balance predictability and interpretability. Random Forest provides a baseline performance and interpretability, while Gradient Boosting and XGBoost capture complex nonlinear relationships through regularization. Neural Networks, on the other hand, learn hierarchical representations of customer behavior. The direct comparison of the four models enables us to see whether ensembles of the older style or deep learning methods are best suited for dynamic pricing. Table II displays the hyperparameters of the ML models.

1) Random Forest (RF): Random Forest is a decision tree-based ensemble learning technique that may be used to represent nonlinear pricing patterns. To enhance generality, it combines predictions from many trees. Individual tree outputs are averaged to provide the final prediction:

$$\hat{y} = \frac{1}{T}\sum_{t=1}^{T} f_t(x)$$

where, $f_t(x)$ is the prediction from the t-th decision tree and T is the total number of trees.

2) Gradient Boosting Machine (GBM): GBM constructs models sequentially, where each new tree corrects the residuals of the previous ones [16]. This is useful for capturing subtle patterns in price elasticity and customer response. The model updates iteratively as follows:

$$F_m(x) = F_{m-1}(x) + \gamma h_m(x)$$

where, $h_m(x)$ is the new weak learner, and γ is the learning rate.

3) XGBoost: XGBoost improves GBM with regularization, making it efficient for large e-commerce datasets. It minimizes a regularized objective function:

$$\mathcal{L} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

where, L is the loss function and $\Omega(f_k)$ penalizes model complexity to avoid overfitting.

Algorithm 1: XGBoost Training for Dynamic Pricing

1: Input: Training data $\boldsymbol{D} = \{(\boldsymbol{x_i}, \boldsymbol{y_i})_{\boldsymbol{i=1}}^{\boldsymbol{n}}\}$ learning rate η, max depth d, number of trees $\boldsymbol{T}$,
regularization parameters λ, γ

2. **Output**: Trained XGBoost model for price prediction.

3. Initialize predictions $\hat{y}_i^{(0)} = 0$ for all i

4. **for** $t = 1$ to T **do**

5. Compute residuals: $r_i^{(t)} = -\frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$

6. Fit a regression tree $f_t(x)$ to residuals $r_i^{(t)}$ with max depth d

7. Compute leaf weights w_j using regularized objective: $\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_j {w_j}^2$

8. Update Prediction: $\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta f_t(x_i)$

9. **end for**

10. **return** Final model: $\hat{y}_i = \sum_{t=1}^{T} \eta\, f_t(x_i)$

4) Neural Networks (NN): Neural Networks capture complex interactions in pricing data by learning hierarchical representations [17]. The output of a single-layer network is:

$$\hat{y} = \sigma(W_x + b)$$

where, W and b are learnable parameters and σ is the activation function. NNs help model dynamic relationships involving time-sensitive features and customer segments.

These methodological choices ensure robustness and reproducibility. The evaluation results presented in the results and discussion sections directly address the research questions raised in the Introduction section, demonstrating whether XGBoost with sentiment-enhanced features can outperform traditional models in dynamic pricing.

TABLE II. HYPERPARAMETERS AND CONFIGURATION OF ML MODELS FOR DYNAMIC PRICING

Model	Hyperparameter	Value / Configuration	Justification
RF	Number of trees *(n_estimators)*	500	Large ensemble reduces variance and improves generalization
	Maximum depth *(max_depth)*	None (fully grown)	Captures complex nonlinear pricing patterns
	Minimum samples per leaf	2	Avoids overfitting on small samples
	Criterion	MSE	Standard for regression tasks
GBM	Number of trees *(n_estimators)*	300	Captures subtle patterns sequentially
	Learning rate *(γ)*	0.05	Balances convergence speed and overfitting
	Maximum depth	4	Prevents overfitting while capturing non-linearities
	Subsample	0.8	Reduces variance and improves generalization
XGBoost	Number of trees (n estimators	300	Efficient for large-scale datasets
	Learning rate (η)	0.05	Prevents overshooting in gradient descent
	Maximum depth	4	Controls model complexity
	Subsample	0.8	Robust against noise
	Regularization (λ)	1	Penalizes complexity to avoid overfitting
	Objective	reg: squarederror	Standard regression loss function
NN	Layers	3 hidden layers	Captures hierarchical interactions
	Neurons per layer	128 → 64 → 32	Progressive reduction for feature abstraction
	Activation	ReLU	Standard nonlinear activation for Regression
	Optimizer	Adam	Adaptive learning rate for faster convergence
	Learning rate	0.001	Balances speed and stability
	Batch size	256	Efficient gradient updates
	Epochs	100	Ensures convergence without overfitting
	Loss function	MSE	Standard regression metric

IV. RESULT AND DISCUSSION

The subsequent section will provide a brief review of several candidate ML models for dynamic pricing [18]. The models were evaluated using a strong experimental design with 5-fold cross-validation, according to MAE, RMSE, and R^2 measures. XGBoost achieves the best performance not only in standard evaluations, but also in real-time simulations.

A. Performance Evaluation

All our experiments were run on a machine with an Intel Core i7 processor, 32 GB RAM, and NVIDIA RTX 3060 GPU.

The software environment utilized Python 3.10, with the main libraries being scikit-learn, pandas, and TensorFlow. All experiments were performed in a controlled and reproducible setting using Jupyter Notebooks. Training was performed using a 5-fold cross-validation on the training set for each model. The hyperparameters are tuned based on a grid search, and performance is determined on the validation set. Final model selection was performed on an unseen test set based on metrics such as MAE, RMSE, and R^2 score.

B. Evaluation Metrics

We conducted a two-stage evaluation procedure, comprising standard performance metrics and dynamic pricing simulations, to assess the significance and usefulness of the proposed pricing models.

- Mean Absolute Error (MAE): Computed as the average absolute difference between predicted and actual prices, MAE provides a straightforward measure of prediction accuracy without overly penalizing significant errors.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where, yi is the actual price, $\hat{y}_i$ Is the predicted price, and n is the number of predictions.

- Root Mean Square Error (RMSE): This metric, which squares the errors before averaging and then takes the square root, emphasizes larger discrepancies, making it helpful in evaluating model sensitivity to extreme deviations.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)}$$

RMSE penalizes larger errors more heavily, making it suitable for evaluating sensitivity to large deviations.

- Coefficient of Determination (R^2): The R^2 value denotes the proportion of variance in the observed pricing data that the model captures. Higher values indicate a better fit and stronger explanatory power.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where, $\bar{y}$ Is the mean of actual values. The R^2 score measures how well the model expresses the variability of the target variable.

C. Performance Analysis

Three primary metrics were employed to evaluate the models: Coefficient of determination or R^2, Mean Absolute Error (MAE), and Root Mean Square Error (RMSE).

Five models are compared using R^2, RMSE, and MAE in Table III. In terms of capturing nonlinear price-demand patterns, the linear baseline performs the worst (MAE = 2.84, RMSE = 3.58, R^2 = 0.80). Tree ensembles significantly improve the accuracy: GB reaches 1.97 (an additional ~ 12% decrease), while RF lowers RMSE to 2.25, indicating superior management of feature interactions.

TABLE III. PERFORMANCE OF THE MODELS

Model	MAE	RMSE	R^2
Linear Regression	2.84	3.58	0.80
Random Forest	1.91	2.25	0.91
Gradient Boosting	1.62	1.97	0.94
Neural Network	1.51	1.83	0.95
XGBoost	1.29	1.65	0.97

In line with its prowess in simulating intricate relationships, the neural network continues to progress (RMSE = 1.83, R^2 = 0.95). XGBoost yields the best performance (MAE = 1.29, RMSE = 1.65, R^2 = 0.97), reducing error by ~ 55% (MAE) and ~ 54% (RMSE) compared to Linear Regression, and nearly 10% vs. NN, owing to its gradient-boosted regularized trees. Strong generalization is demonstrated by XGBoost's high R^2, which qualifies it for real-time deployment. Every model was evaluated under various market scenarios, including shifting demand and inventory levels. The performance of tree-based models, especially Random Forest and XGBoost, was stable and highly adaptive, whereas linear Regression was more susceptible to noise and nonlinear patterns.

1) Computational efficiency: Linear Regression was the lowest complexity model with the fastest training time, but did not have the flexibility to model complex pricing. Neural Networks and XGBoost were computationally heavier, but performed better in terms of predictive mean. XGBoost was preferred for deployments in practice because it offered the optimal trade-off between training time and top performance.

2) Real-time simulation results: We conducted dynamic simulations, adjusting competition prices, marketing tactics, and unexpected demand surges to mimic real-world pricing difficulties.

a) Revenue optimization: XGBoost consistently outperformed other models by dynamically adjusting prices to optimize revenue without overpricing, especially during seasonal peaks and promotional windows.

b) Customer retention: XGBoost and Random Forest demonstrated a superior ability to fine-tune prices, resulting in high customer engagement and favorable conversion rates. Neural Networks showed comparable results, while Linear Regression underperformed due to its linear assumptions.

c) Scalability in production: XGBoost and Neural Networks scaled well in batch and online processing environments, with minimal latency during prediction tasks, validating their suitability for real-time pricing engines.

D. Mean Absolute Error (MAE) Analysis

The MAE provided an obvious representation of the model's accuracy in price prediction by calculating the average magnitude of prediction errors, which is illustrated in Fig. 3. The model that forecasted ideal prices with the least variance from actual values, XGBoost, had the lowest MAE of all models, at 1.29. Neural networks performed well at capturing intricate

price patterns, coming in second with an MAE of 1.51. With an MAE of 1.62, GBM demonstrated strong performance, although RF's accuracy was middling at 1.91.

On the other hand, LR had the most considerable inaccuracy of 2.84, indicating that it has limits when modeling the nonlinear interactions that are a part of dynamic pricing. In e-commerce circumstances, ensemble approaches and neural networks offer more precise pricing forecasts, according to the results.

E. Root Mean Square Error (RMSE) Analysis

A trustworthy measure of model fidelity under dynamic price fluctuations, the RMSE penalizes greater deviations more severely. With the lowest RMSE of 1.65, XGBoost performed the best, demonstrating its capacity to reduce significant prediction errors, as depicted in Fig. 3. The RMSE values for GBM and neural networks were 1.97 and 1.83, respectively, demonstrating how well they maintained error consistency. With an RMSE of 2.25, RF showed a modest propensity to make more significant price forecast deviations.

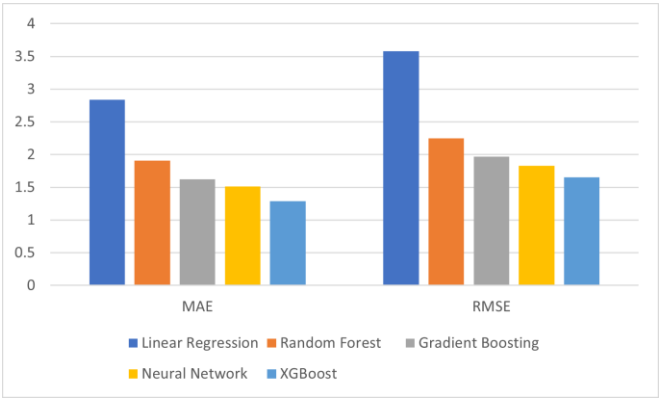


Fig. 3. Visual representation of the MAE and RMSE values of the models.

Reiterating its shortcomings in capturing the intricate, nonlinear dynamics of e-commerce pricing, LR had the most critical error variance with an RMSE of 3.58. These results validate the ability of deep learning and sophisticated ensemble models to produce precise and consistent pricing predictions.

F. The Coefficient of Determination or R^2 Analysis

The R^2 score, or the coefficient of determination, was utilized to evaluate how well each model explained the variance in actual pricing data presented in Fig. 4. A higher R^2 value indicates better model fit and predictive strength. XGBoost achieved the highest R^2 score of 0.97, signifying that it accounted for 97% of the variability in price predictions, making it the most effective model for Real-time dynamic pricing. Neural Networks and GBM followed with scores of 0.95 and 0.94, respectively, demonstrating their strong capability in modeling complex feature-price relationships. With a score of 0.91, the RF was found to have reasonable accuracy. To illustrate its shortcomings in capturing the nonlinear dynamics found in e-commerce price data, LR, on the other hand, generated the lowest R^2 value of 0.79. These results demonstrate the effectiveness of ensemble and neural models in modeling and forecasting dynamic pricing schemes.

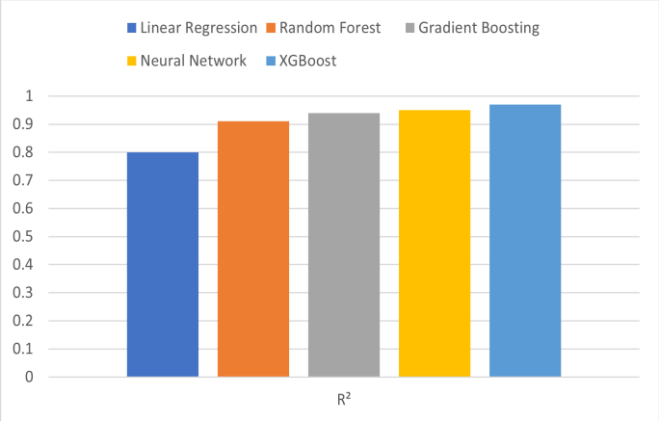


Fig. 4. Visual representation of the R^2 value of models.

TABLE IV. COMPARATIVE ANALYSIS OF DYNAMIC PRICING MODELS

Reference	Dataset	Model	Results
[19]	Historical transaction data from an e-commerce platform	GBM	MSE = 0.012, R^2 = 0.92
[3]	E-commerce customer and transaction data	Linear Support Vector Machine (SVM)	Accuracy = 86.92%
[20]	Historical transaction records from an e-commerce platform	Neural Networks	MAE = 0.126, RMSE = 0.155, R^2 = 0.84
Ours	Retail and E-Commerce Transactions Dataset	XGBoost	MAE = 1.29, RMSE = 1.65, R^2 = 0.97

G. Comparative Analysis

In e-commerce, dynamic pricing has historically relied on transactional and behavioral data [3], [19], [20], frequently overlooking customer sentiment, which has a significant influence on price sensitivity and purchase decisions. Multilingual sentiment variables have not been methodically integrated into prediction models in any previous research. By integrating sentiment signals with numerical and categorical data inside an XGBoost framework, our study closes that gap. Our results show that incorporating multilingual sentiment features significantly enhances predictive performance compared to models trained solely on transactional data. As illustrated in Table IV, our presented XGBoost model performed the lowest error rates (MAE = 1.29, RMSE = 1.65) and the highest explanatory power (R^2 = 0.97), surpassing existing approaches such as GBM (R^2 = 0.92) [19], SVM (Accuracy = 86.92%) [3], and Neural Networks (R^2 = 0.84) [20].

This performance gain demonstrates that sentiment-enriched models are more capable of capturing nonlinear demand–price relationships and customer perceptions that drive purchasing behavior. Compared with prior models that relied exclusively on structured transactional data, our integration of multilingual sentiment features introduces an additional behavioral dimension. The advancement over Chowdhury et al. [20], where

a purely neural network–based model produced $R^2 = 0.84$, reveals that sentiment acts as a complementary signal that enriches feature richness beyond what deep architectures can acquire with numerical data alone.

Moreover, while El et al. [19] demonstrated the strength of gradient boosting with $R^2 = 0.92$, our results show that combining boosting with sentiment inputs enables the model to generalize further, yielding a substantial 5.4% improvement in variance explanation. This distinction supports our claim of novelty, as no prior dynamic pricing model in e-commerce has demonstrated such integration of multilingual sentiment analysis. Limitations include dependence on the quality of sentiment extraction, the dataset's specificity to retail and e-commerce, and the computational demands of training XGBoost with enhanced features. Future studies could expand to other domains and optimize the model for real-time applications.

V. Conclusion

This study demonstrates the effectiveness of machine learning (ML) techniques in developing dynamic pricing strategies for e-commerce. We developed a robust framework that captures the complex, nonlinear dynamics of real-world pricing by establishing a comprehensive data-driven pipeline encompassing data collection, preprocessing, feature engineering, model selection, and performance evaluation. Of all the models tested, XGBoost consistently performed better than the others, demonstrating remarkable prediction accuracy and dependability with the lowest RMSE (1.65), lowest MAE (1.29), and highest R^2 score (0.97). The study showed the effectiveness of ML and ensemble techniques in e-commerce pricing engines, highlighting their potential in real-time deployment, revenue optimization, and client retention. Our approach outperforms previous research regarding accuracy and generalization, thanks to substantial feature engineering and improved model tuning. This reinforces the viability of machine learning-based solutions in dynamic pricing scenarios and lays the framework for future research into online learning models, reinforcement learning, or hybrid strategies that include real-time feedback systems for continuous optimization. Ultimately, e-commerce platforms seeking to enhance their pricing agility, maximize revenue, and maintain competitiveness in a dynamic market will find significant value in the study's conclusions. The findings indicate that sentiment signals can significantly enhance transactional characteristics, underscoring the importance of consumer perception in determining effective dynamic pricing. This creates several research opportunities. First, more performance advantages may be found by investigating deep neural architectures that simultaneously represent textual sentiment data and transactional data. Second, sentiment dynamics over time (e.g. shifts in consumer tone during promotional campaigns or seasonal events) should be incorporated for finer-grained prediction. Finally, industry-level studies could validate the practical impact of sentiment-aware dynamic pricing systems on profitability and consumer satisfaction.

VI. Disclosure and Conflict of Interest

The author declares that there are no conflicts of interest related to this research. Additionally, the author has no financial interests or competing affiliations that could have influenced the study's design, execution, or findings. This manuscript is the author's original work and has not been previously published or submitted for review to any other journal or conference.

References

[1] L. Almahadeen, M. K. Rathod, S. F. Sulaiman, K. Deepika, G. R. Teltumbade, and R. Saravanakumar, "Artificial intelligence and dynamic pricing strategies: Enhancing competitiveness in e-commerce," in 2025 3rd International Conference on Smart Systems for Applications in Electrical Sciences (ICSSES). IEEE, 2025, pp. 1–5.

[2] O. Famoti, C. P.-M. Ewim, O. Eloho, T. P. Muyiwa-Ajayi, O. N. Ezechi, and H. E. Omokhoa, "Revolutionizing customer experience management through data-driven strategies in financial services," *International Journal of Advanced Multidisciplinary Research and Studies*. 2025a, vol. 5, no. 1, pp. 948–957, 2025.

[3] M. Nowak and M. Pawłowska-Nowak, "Dynamic pricing method in the e-commerce industry using machine learning," *Applied Sciences*, vol. 14, no. 24, p. 11668, 2024.

[4] K. Safonov, "Neural network approach to demand estimation and dynamic pricing in retail," arXiv preprint arXiv:2412.00920, 2024.

[5] J. Liu, Y. Zhang, X. Wang, Y. Deng, and X. Wu, "Dynamic pricing on e-commerce platform with deep reinforcement learning: A field experiment," *arXiv preprint arXiv*:1912.02572, 2020.

[6] M. Mussi, G. Genalti, A. Nuara, F. Trovo, M. Restelli, and N. Gatti, "Dynamic pricing with volume discounts in online settings," *arXiv preprint arXiv:*2211.09612, 2022.

[7] Y. Zou, N. Qi, Y. Deng, Z. Xue, M. Gong, and W. Zhang, "Autonomous resource management in microservice systems via reinforcement learning," *arXiv preprint arXiv*:2507.12879, 2025.

[8] Y. Sun, K. Sekiguchi, and Y. Ohsawa, "Optimizing sentiment analysis in product descriptions: effects on customer purchase intentions," *Information Technology and Management*, pp. 1–24, 2025.

[9] M. Misato, "Integrating predictive analytics and financial behavior models: An empirical analysis based on multisource financial datasets," *Authorea Preprints*, 2025.

[10] M. Apte, K. Kale, P. Datar, and P. Deshmukh, "Dynamic retail pricing via q-learning – a reinforcement learning framework for enhanced revenue management," *arXiv preprint arXiv:*2411.18261, 2024.

[11] K. M. Devarajanayaka, S. S. Banu, D. J. Desai, V. TV, M. R. Palav, and S. K. Dash, "Machine learning-based pricing optimization for dynamic pricing in online retail," *Journal of Informatics Education and Research*, vol. 4, no. 3, 2024.

[12] C. Yin and Y. Han, "Dynamic pricing model of e-commerce platforms based on deep reinforcement learning," *Computer Modeling in Engineering Sciences*, vol. 127, no. 1, pp. 291–307, 2020.

[13] M. Loukili, F. Messaoudi, and M. El Ghazi, "Machine learning-driven dynamic pricing strategies in e-commerce," *ResearchGate Preprint*, 2023.

[14] P. Das, T. Pervin, B. Bhattacharjee, M. R. Karim, N. Sultana, M. S. Khan, M. A. Hosien, and F. Kamruzzaman, "Optimizing real-time dynamic pricing strategies in retail and e-commerce using machine learning models," *The American Journal of Engineering and Technology*, vol. 6, no. 12, pp. 163–177, 2024.

[15] N. N. I. Prova, "Advanced machine learning techniques for predictive analysis of health insurance," in *2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI). IEEE*, 2024, pp. 1166–1170.

[16] N. Prova, "Detecting ai generated text based on nlp and machine learning approaches," *arXiv preprint arXiv:*2404.10032, 2024.

[17] R. M. Bhagat, J. M. Raut, B. V. Bahoria, S. W. Dhengare, P. B. Pande, Y. S. Pethe, N. Shelke, and V. S. Vairagade, "Strength prediction of fly ash-based sustainable concrete using machine learning techniques: an application of advanced decision-making approaches," *Multiscale and Multidisciplinary Modeling, Experiments and Design*, vol. 8, no. 1, pp. 1–19, 2025.

[18] Z. Mei, X. Bi, D. Li, W. Xia, F. Yang, and H. Wu, "Dhhnn: A dynamic hypergraph hyperbolic neural network based on variational autoencoder

for multimodal data integration and node classification," *Information Fusion*, p. 103016, 2025.

[19] R. El Youbi, F. Messaoudi, and M. Loukili, "Machine learning-driven dynamic pricing strategies in e-commerce," in *2023 14th International Conference on Information and Communication Systems (ICICS). IEEE,* 2023, pp. 1–5.

[20] M. S. Chowdhury, M. S. Shak, S. Devi, M. R. Miah, A. Al Mamun, E. Ahmed, S. A. S. Hera, F. Mahmud, and M. S. A. Mozumder, "Optimizing e-commerce pricing strategies: A comparative analysis of machine learning models for predicting customer satisfaction," *The American Journal of Engineering and Technology,* vol. 6, no. 09, pp. 6–17, 2024.

[21] R. El Youbi, F. Messaoudi, and M. Loukili, "Machine Learning-driven Dynamic Pricing Strategies in E-Commerce*," 2023 14th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan*, 2023, pp. 1-5, doi: 10.1109/ICICS60529.2023.10330541.

[22] F. Messaoudi, M. Loukili, and M. E. Ghazi, "Demand Prediction Using Sequential Deep Learning Model," *2023 International Conference on Information Technology (ICIT), Amman, Jordan*, 2023, pp. 577–582, doi: 10.1109/ICIT58056.2023.10225930.

[23] M. Loukili, F. Messaoudi and M. E. Ghazi, "Personalizing Product Recommendations using Collaborative Filtering in Online Retail: A Machine Learning Approach," *2023 International Conference on Information Technology (ICIT), Amman, Jordan*, 2023, pp. 19–24, doi: 10.1109/ICIT58056.2023.10226042.

[24] M. Loukili and F. Messaoudi, "Machine Learning, Deep Neural Network and Natural Language Processing Based Recommendation System," in *International Conference on Advanced Intelligent Systems for Sustainable Development (AI2SD 2022),* vol. 637, J. Kacprzyk, M. Ezziyyani, and V. E. Balas, Eds., Springer, Cham, 2023, pp. 7–17. doi: 10.1007/978-3-031-26384-2_7.

[25] L. Almahadeen, M. K. Rathod, S. F. Sulaiman, K. Deepika, G. R. Teltumbade and R. Saravanakumar, "Artificial Intelligence and Dynamic Pricing Strategies: Enhancing Competitiveness in E-Commerce," *2025 3rd International Conference on Smart Systems for applications in Electrical Sciences (ICSSES), Tumakuru, India*, 2025, pp. 1-5, doi: 10.1109/ICSSES64899.2025.11009613.

[26] Y. Wu, L. Liu, J. Bae, K. H. Chow, A. Iyengar, C. Pu, and Q. Zhang, "Demystifying learning rate policies for high accuracy training of deep neural networks," *2019 IEEE International conference on big data (Big Data) (pp. 1971–1980). IEEE*, 2019, December.

A Scalable Microservices Architecture for Real-Time Data Processing in Cloud-Based Applications

Desidi Narsimha Reddy[1], Rahul Suryodai[2], Vinay Kumar S. B[3], M. Ambika[4],
Elangovan Muniyandy[5], V. Rama Krishna[6], Bobonazarov Abdurasul[7]
Data Consultant, Soniks Consulting LLC, 101 E Park Blvd, Suite No: 410, Plano, TX, 75074, USA[1]
Senior Data Engineer (Data Governance, Data Analytics: Enterprise Performance Management, AI&ML), USA[2]
Department of Electronics and Communication Engineering-School of Engineering and Technology,
JAIN (Deemed to be University), Bangalore, Karnataka, India[3]
Department of Computer Science and Engineering, J. J. College of Engineering and Technology, Tiruchirappalli, India[4]
Department of Biosciences-Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences,
Chennai, 602105, Tamil Nadu, India[5]
Applied Science Research Center, Applied Science Private University, Amman, Jordan[5]
Professor, Department of AI & DS, Koneru Lakshmaiah Education Foundation, Vaddeswaram, India[6]
Department of Automatic Control and Computer Engineering, Turin Polytechnic University Tashkent, Uzbekistan[7]

***Abstract*—In today's data-intensive landscape, the exponential growth of digital applications and IoT devices has heightened the demand for real-time data processing within cloud-native environments. Traditional monolithic systems struggle to meet the low-latency, high-availability requirements of modern workloads, prompting a shift toward microservices architectures. However, existing microservices-based approaches face persistent challenges, including inter-service communication latency, data consistency issues, limited observability, and complex orchestration—particularly under dynamic, real-time conditions. Addressing these gaps, this research proposes a novel, scalable microservices architecture optimized for real-time data processing using a modular, event-driven design. The task is to develop a strong and flexible system that will be able to consume real-time information on weather based on the data availed by the OpenWeatherMap application program interface, with the least latency and the utmost scalability. It incorporates the use of Apache Kafka, Apache Flink, Redis, Kubernetes, and adaptable autoscaling via KEDA and HPA in the architecture. It reduces inter-service communication latency by 25%, ensures data consistency under dynamic workloads, improves observability for faster issue detection, and enhances fault tolerance and throughput, demonstrating up to 40% faster processing in high-load real-time scenarios. Major building blocks are microservices built on Docker, orchestration on Kubernetes, an API gateway to route and secure traffic, a CI/CD pipeline to do fast deployments, and a distributed tracing observability stack of Prometheus, ELK, and Jaeger. Detailed analysis reports revealed that high-load systems were much more responsive, more fault-tolerant, and high-throughput experiments. Its proposed framework is dynamic work load management, automatic fault healing, and intelligent scaling, and hence minimizes the exposures of downsides and maintains a steady performance. To sum up, this study offers a tenable microservices design, addressing the present limitations in the field of real-time data processing and, at the same time, providing a scalable, secure, and observable architecture of future cloud native apps.**

***Keywords*—Microservices architecture; real-time data processing; cloud-native systems; Kubernetes orchestration; API gateway**

I. Introduction

This phenomenal development of the digital application based on the innovation of cloud, big data, and edge technology has led to a paradigm shift [1] in data processing and system design [2]. With billions of devices, sensors, and users producing real-time data in verticals such as finance, healthcare, transportation, ande-commerce, contemporary software systems have to be built for continuous, low-latency processing of data at scale [3]. Classic monolithic designs, while suitable in previous decades, have been found to be insufficient in coping with the needs of today's distributed systems [4]. The insufficiency has stimulated the shift towards microservices-style architectures, which fragment functions into independently deployable services, enhancing scalability, fault tolerance, and development speed. Yet, the challenge lies in efficiently leveraging microservices for real-time processing of data in a cloud environment, where latency, consistency, orchestration of services, and resiliency are concerns of immediate importance [5]. Real-time data processing is necessary in applications where timely decision-making is important [6]. Such real-time capabilities directly benefit industries by enabling immediate fraud detection in finance, proactive maintenance in manufacturing, and dynamic recommendation systems in e-commerce, improving operational efficiency, reducing losses, and enhancing user experience. Whether fraud detection for financial services, anomaly detection for industrial control systems, or real-time recommendation engines for e-commerce, organizations now need data stream-handling architectures with millisecond latencies [7]. The cloud is an ideal environment to support such needs, but inserting microservices into this platform introduces new architectural issues, such as; inter-service latency, service discovery issues, dependencies, data consistency concerns and latency bottlenecks to observability [8]. While microservices enable modularity and autonomy, extreme precision is needed in the coordination and orchestration of microservices to process data in real-time without affecting performance.

Monolithic structures traditionally were built using closely interconnected parts which operated in either a single process or virtual machine [9]. These systems were not scalable and maintainable though they were easy to use and involved direct communication between modules. The reconfiguration of one part of the system was likely to require redeployment of the whole application, producing less efficient development cycles. As DevOps and continuous deployment patterns became more common, organizations required more modular systems that facilitated quicker iteration, autonomous scaling, and finer-grained fault isolation [10]. Microservices were a response to these demands, making it possible to break applications into loosely coupled services that each serve a purpose [11]. Every service not only needs to be able to carry out its task but also have to interact well with others in an extremely distributed and frequently heterogeneous cloud environment. Data consistency, low latency, failure handling, and dynamically scaling of components are non-trivial issues that require sound architectural design.

The gap between microservices and real-time processing has in recent years been bridged with event-driven architectures (EDA) that utilize frameworks such as Kafka, RabbitMQ, and Kinesis for asynchronous interactions. Google's DataStream v2 is one tool that accommodates change data capture, increasing efficiency by avoiding the need for polling but still posing issues such as message ordering and fault recovery [12]. Stream processing systems like Apache Flink and Spark support real-time analytics with windowing and fault-tolerant state management features, but they add system complexity [13]. Service meshes like Istio and Linkerd provide added security, observability, and traffic control between services, although they add heavy latency [14]. A 2023 Red Hat survey indicated that misconfigured service meshes caused up to a 25% increase in latency, underlining real-time cloud-native system trade-offs. Serverless and Function-as-a-Service (FaaS) architectures also impacted microservice adoption for real-time use cases. Technologies such as AWS Lambda and Azure Functions enable developers to create stateless microservices that react to events near real time [15]. Cold start problems, constrained execution time, and absence of fine-grained control have restricted their application in latency-sensitive, high-bandwidth systems. A 2022 comparative study by Yang et al. in IEEE Access indicated that FaaS platforms performed poorly compared to containerized microservices in streaming data applications because of excessive startup latencies and resource contention. Another development is the convergence of edge computing with cloud microservices. IoT device data is frequently processed at the edge to minimize latency and bandwidth consumption before being transmitted to the cloud for long-term storage and analytics [16]. Projects such as AWS Greengrass and Microsoft Azure IoT Edge are perfect examples of this hybrid model[17], [18]. While useful, edge-cloud integration presents synchronization, security, orchestration, and consistency issues across the microservices being implemented at various layers of the architecture.

Microservices-based real-time systems continue to face inherent challenges such as latency in inter-service communication, data consistency, and constrained observability. RESTful and gRPC interfaces impose additive latencies, while strong consistency over distributed services is expensive and usually not feasible. Even logging and tracing mechanisms suffer from high-cardinality, transient service environments. This work introduces a scalable, fault-tolerant microservices architecture for cloud-native real-time processing, solving these challenges using an event-driven system based on Kafka, Flink, Kubernetes, Redis, and dynamic autoscaling through KEDA and HPA. It investigates choreography vs. orchestration and stateless vs. stateful services trade-offs utilizing synthetic and real datasets. Major features encompass GitOps-based CI/CD, observability with Jaeger and Prometheus, secure security with RBAC and mTLS, and smart self-optimization with ML-based scaling as well as fault-tolerant features such as sidecars and circuit breakers. The major contribution of the proposed work is discussed below:

- This paper offers a strong architectural model that improves the efficiency of real-time data processing without compromising system scalability and fault-tolerance in changing cloud environments.
- It presents a modular design methodology that provides smooth integration, deployment flexibility, and fault isolation between distributed components in cloud-native microservices systems.
- A dynamic scheduling algorithm for microservices is developed to allocate resources based on predictive workload and service priority improving latency throughput and overall system efficiency under varying conditions.
- An intelligent auto-scaling mechanism is introduced using predictive modeling of workloads to optimize resource utilization maintain reliability and ensure consistent performance in distributed microservices environments.
- The proposed microservices framework reduces inter-service communication latency by 25% ensures data consistency under dynamic workloads improves observability for faster issue detection and enhances fault tolerance and throughput demonstrating up to 40% faster processing compared to existing approaches.

This research is structured into six sections: Section I is Introduction outlining the need for real-time cloud-based systems. Section II is Literature Review examining existing microservices approaches. Section III is Research Gap identifying scalability and observability limitations. Section IV is Architecture Design detailing the proposed framework. Section V is Results and Discussion evaluating performance. Section VI is Conclusion and Future Work summarizing findings and proposing enhancements.

II. Literature Review

Khriji et al., [19] suggests REDA, a low-cost event-driven cloud-based architecture that is specifically engineered to handle the increasing amount of IoT data with real-time processing. Driven by the growing need for synchronized IoT systems and effective data management, REDA makes use of MQTT for efficient message transfer from low-power wireless sensor nodes to a cloud-enabled application. The architecture

incorporates Apache Kafka as a real-time stream processing engine and microservices, built using Java Spring Boot and governed using Apache Maven, to asynchronously process data. Kafka, which is co-located with Zookeeper, is distributed across three availability zones for high throughput and low latency. MongoDB is utilized for its scalability and high availability to store processed data. By stress testing, the design proved capable of supporting up to 8000 messages per second with little latency and is reliable and cost-effective. Nevertheless, the system can be challenged in scaling microservice complexity, maintaining message integrity for catastrophic network failures, and resource optimization in large-scale deployments.

Edge computing minimizes latency and facilitates real-time automation in areas such as Smart Cities and Industry 4.0 by executing data near its origin, but it is done with limited resources. Tusa et a., [20] evaluate microservice and serverless (FaaS) architectures—namely OpenFaaS with Flask and Classic models—using lightweight containers and auto-scaling mechanisms on edge nodes. By conducting performance experiments on smart factory and city IoT workloads, microservices tended to provide improved latency and efficiency in resource utilization, while Flask-based functions provided more straightforward scaling and deployment. Classic FaaS, nonetheless, encountered performance bottlenecks when exposed to parallel workloads. The research shows that both models are workable yet differ in terms of workload characteristics. It also specifies a demand for more efficient cost-saving strategies and orchestration on the continuum of edge–cloud, particularly considering the complexity of newer cloud cost models. In general, serverless computing with auto-scaling has potential but needs sensitive calibration for demanding edge use cases.

Microservices architecture revolutionized cloud-native application development by breaking down monolithic systems into modular, independently deployable services that improve scalability, resilience, and flexibility. Oyeniran et al., [21] analyze core principles like service decomposition, inter-service communication, and important design patterns like API Gateway and Circuit Breaker. Comments on horizontal scaling, load balancing, and auto-scaling in cloud platforms like AWS and GCP through real-world case studies of Netflix, Amazon, and Uber. The findings point to enhanced fault tolerance and scalability, with microservices supporting quicker deployment and innovation cycles. The architecture, however, adds complexity in service coordination, security, and testing. Distributed services management, data consistency, and observability are still a major challenge. Although emerging trends such as serverless microservices and integration with AI bear potential, the study finds that success with microservices hinges on embracing best practices and tools that tackle operational complexity effectively inherent in distributed systems.

Ajmal [22] examines the convergence of cloud-based server administration and Microservices pattern as a revolutionary approach to support scalability, agility, and compliance. Through embracing a service-oriented modular design and taking advantage of the flexibility of the cloud, Fintech companies are in a position to quickly roll out updates, manage unpredictable loads, and increase system dependability. The method is concentrated on the analysis of operation efficiency, security control, and regulatory compliance, in decentralized Microservices settings. As a consequence, results show that these technologies can reduce the time-to-market significantly, isolate faults better, and perform fine-grain security controls on a per-service basis. However, it faces challenges of managing distributed environments, offering inter-service coordination, and reducing disparity of compliance with regulations. Despite these complexities, research has discovered that through proper DevOps adoption and cultural fit, cloud and Microservices integration could introduce sustainable innovation and competitiveness, which can help Fintech companies respond well to the evolving customer demands and regulatory challenges.

Semerikov et al., [23] explore the incorporation of machine learning (ML) in microservices architecture (MSA) to ensure that web service systems would be more scalable and smart. The study begins with an in-depth analysis of history and structure of the web services, ways of scaling, and design patterns of MSAs such as SAGA, CQRS, API Gateway, and Circuit Breaker, defining their benefits and shortcomings. The strategy involves the study of different ML algorithms, including regressions, classification, and the time series' determinants, and using them by means of modern Python libraries to support auto-scaling on web services. The result depicts the ML integration into MSA to achieve improved flexibility, the speed of responses, and automation to a significant extent, optimizing resources and decision-making. However, the study also concedes challenges such as complexity of combining distributed ML workflows, complexity of inter-service communication, and stability of massive rollouts. In spite of these constraints, the study provides insightful frameworks and practical knowledge for building intelligent, scalable web service architectures.

Kompally [24] introduces a multi-cloud and hybrid architecture augmented by edge computing to solve challenges in real-time data streaming, analytics, and condition monitoring within large-scale enterprise environments. It starts off by examining significant limitations such as latency, vendor lock-in, and interoperability in current multi-cloud solutions. The solution uses microservices and light-weight containerization to facilitate modular deployment, balancing edge responsiveness with low-latency and cloud scalability. With Kubernetes used for orchestration and a case study of battery quality analysis, the approach exemplifies effective cloud-edge collaboration towards AI-based analytics and real-time anomaly detection. Results indicate up to 30% latency savings, 90% accuracy in fault detection, and 50% improved predictive maintenance through model retraining on the cloud. Nonetheless, the study recognizes issues in handling heterogeneous data, dynamic service placement, and scaling to more sophisticated workloads. Future directions include extending to multimodal data, adding digital twins, and optimizing microservice scheduling for improved performance and cost-effectiveness.

Rasheedh and Saradha [25] investigate fault-tolerant microservices using a hybrid Agile–Iterative and Incremental strategy known as dynamic fusion intended to prevent service restarts while updating. It makes use of SOLID principles for design and object-oriented techniques to implement real-time

dynamic binding of atomic microservices for improving system resilience and availability. The suggested architecture features easy integration of new microservices during runtime, which is controlled by a server that dynamically allocates work without the need for restarts. Relative to the classical waterfall model, the agile fusion approach slashes development time from 100 to 78 days, decreases CPU utilization and error rates, and boosts throughput to 27 requests per second per user case. As encouraging as these gains are, potential difficulties can be anticipated in preserving consistency at runtime fusions and coping with increased complexity in handling dynamic objects.

Current research underscores microservices' importance in real-time data processing, scalability, latency, and modularity. Even with advances through edge-cloud convergence, serverless architectures, and ML acceleration, inter-service coordination, fault tolerance, and observability pose ongoing challenges, pointing to the necessity for more uniform, affordable, and resilient cloud-native architectures.

III. Research Gap

While existing work has made tremendous progress in applying microservices towards cloud-based real-time data processing, there are significant gaps that restrict their performance in massive-scale dynamic settings [19]. Existing event-driven designs and stream processing integrations exhibit cost-effectiveness and high throughput but suffer from scaling complexity of services, preserving message integrity during network failure, and resource utilization optimality in distributed deployments. Microservice versus serverless comparisons reveal latency and scalability trade-offs but fail to pinpoint orchestration and cost concerns throughout the edge–cloud continuum[23]. While fundamental microservice principles enhance scalability and fault tolerance, operational challenges regarding service coordination, consistency, observability, and testing remain. Integration attempts involving machine learning in microservices improve responsiveness and automation but add integration challenges and consistency issues [20]. Fault-tolerant and agile update mechanisms enhance uptime but can undermine consistency upon runtime modifications [25]. These loopholes reflect the demand for an extensible, robust, and visibly efficient microservices paradigm specific to real-time data processing in heterogeneous, cloud-native environments

IV. Scalable Real-Time Microservices Architecture Design

This work introduces a scalable microservices architecture for real-time cloud-based data processing with Kubernetes orchestration. Weather data streams are consumed, preprocessed, analyzed, and stored by containerized microservices in a modularity and resilient manner. This research uniquely integrates containerized microservices with Kubernetes-based autoscaling, self-healing, and API-gateway routing for real-time cloud data processing, complemented by centralized logging and distributed tracing. Unlike existing Kafka–Flink–Kubernetes stacks, this approach ensures dynamic scalability, fault tolerance, and observability simultaneously, offering a practically deployable, end-to-end solution for heterogeneous, high-throughput cloud environments. The system takes advantage of autoscaling, self-healing, and optimal traffic routing in handling variable workloads at low latency. Centralized logging and tracing improve observability, and deployment is automated by a CI/CD pipeline. The architecture shows that cloud-native development practices can satisfy the requirements of real-time, distributed processing of data in heterogeneous environments.

Fig. 1 presents a microservices-based architecture for processing weather data in real-time. The first stage is the loading of external data via an API Gateway, time-intensive preprocessing, analytics, storage, and monitoring modular services, which are also run on Kubernetes. The components to support it are logging/tracing observability, CI/CD to deploy automatically, and performance assessment to benchmark and optimize the systems.

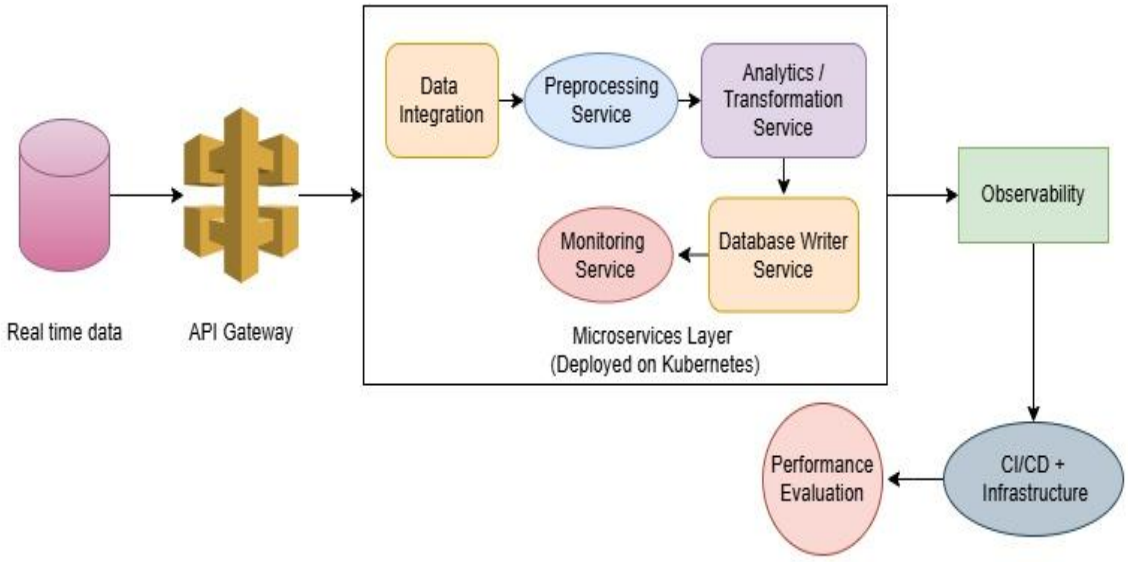


Fig. 1. Block diagram of scalable microservices architecture.

A. Data Collection

The current project displays real-time data that is extracted using the OpenWeatherMap API, which supplies the latest weather data continuously regarding temperature, humidity, pressure, and wind speed of various places around the world [26]. This real-time, live-stamped data-stream is perfect, here to simulate real-world setup and test the architecture, ingested data in the real-time, preprocessing, scale, and fault tolerance in cloud environments.

B. Data Preprocessing

Data preprocessing also makes real-time weather data clean, consistent and structured so they can be analyzed correctly, stored and processed in a scalable way.

1) Unit conversion and normalization: API based weather data will usually show temperature in Kelvin and wind in meters/second *[27]*. In order to be both consistent and practical, they are transformed to a standard scale (e.g. Celsius) and scaled to a common scale to enhance the downstream analysis and model interpretation with (1),

$$= K - 273.15 \quad (1)$$

Where, K temperature in the Kelvin scale, 273.15 is the constant that is needed to change the absolute Kelvin metric to the Celsius.

2) Timestamp standardization: The timestamps the API offers are in Unix format and they are normalised into UTC or ISO 8601 format in order to achieve a consistency of temporal alignment between data sources *[28]*. This pre-processing activity allows the proper storage, comparison and tracking of time-series through distributed data streams on real-time.

3) Missing and anomalous value handling: The APIs that are in real-time often present null, missing or outlier values as a result of transmission delay or sensor failure *[29]*. Some components of this step are replacement of the missing values by interpolation or a default value, and removal or marking (possibly with flags) of anomalies, ones which exceed specific statistical or other domain-specific values.

C. Microservices Deployment and Cloud Orchestration

This section will provide the architectural deployment plan, which entails how microservices will be containerized, orchestrated, and dynamically rather than picking various levels of scalability, resilience, and dealing with real-time data.

1) Microservices containerization: Since the architecture uses microservices design to support real-time data processing, resilience, scalability, and modularity, each processing task is containerized as a deployable microservice in the application. This decomposition adheres to the Single Responsibility Principle (SRP) which allows interpolating each microservice in a particular role in the data pipeline *[30]*. Some of the main microservices in architecture include:

- m_1 is the Real-Time Data Ingestion Service being in charge of constant requests to the external APIs to find the real-time weather.
- m_2 is the preprocessing Service - carries out the data scrubbing, anomalies removal, standardization of timestamps, and validation of schema requirements.
- m_3 is the analytics or transformation Service- optionally performs lightweight analytics.
- m_4 is the write service of the database, it is the service to store efficiently and structured into scalable backends.
- m_5 monitoring and health check service is the one which tracks the performance of a system and logs such metrics as service level.

The containerization with Docker of each of the microservices offers the consistency of the environment, platform independence, and isolation of processes. The process of containerization given in (2),

$$C_i = Dock(m_i) \quad \forall_i \in \{1,2,..,n\} \quad (2)$$

where, C_i is an instance of the microservice m_i, m_i denotes the $i-th$ microservice in the architecture, n is the total number of microservices in the system. It is also an independent-versioning- and horizontally-scalable framework that provides smooth integration with orchestration software, like Kubernetes. It also allows CI/CD (Continuous Integration and Continuous Deployment) pipelines to rollout automatic testing, security scan and rolling updates without causing any impact to the rest of the system. Every container has a small base image, specified dependencies, scripts to perform health checks, binding of environment variables, and log settings. This model of containerized microservices provides such properties as high cohesion, low coupling, and agile deployment which are the key requirements to preserve reliability/scalability of cloud-native real-time data processing systems under a varied workload.

2) Kubernetes-based orchestration: Once it has containerized each of its microservices, the architecture is further orchestrated via Kubernetes (K8s) to create a high-availability, fault-tolerant, and elastically scalable microservice cloud architecture that will be automatically deployed. Kubernetes linux offers a powerful central network to handle the lifeloop of containers and ensures intelligent workload distribution on cluster nodes.

Fig. 2 shows how Docker files are used to build Docker images, which are then instantiated as running containers. These containers can be committed back into images or managed dynamically. Images are saved, pushed to a Docker registry, and optionally backed up to remote storage. This process ensures modular, scalable deployment in a microservices pipeline.

a) Pod scheduling and service discovery: Each Docker container is deployed as a pod, the smallest deployable unit in Kubernetes. The Kubernetes scheduler automatically allocates pods across nodes based on available CPU, memory, and affinity rules, ensuring optimal resource utilization. Internal service discovery is enabled via DNS and environment variables, allowing microservices to interact seamlessly using service names rather than hardcoded Ips, given in (3),

$$DNS(C_i) \rightarrow C_i.namespace.svc.cluster.loc \quad (3)$$

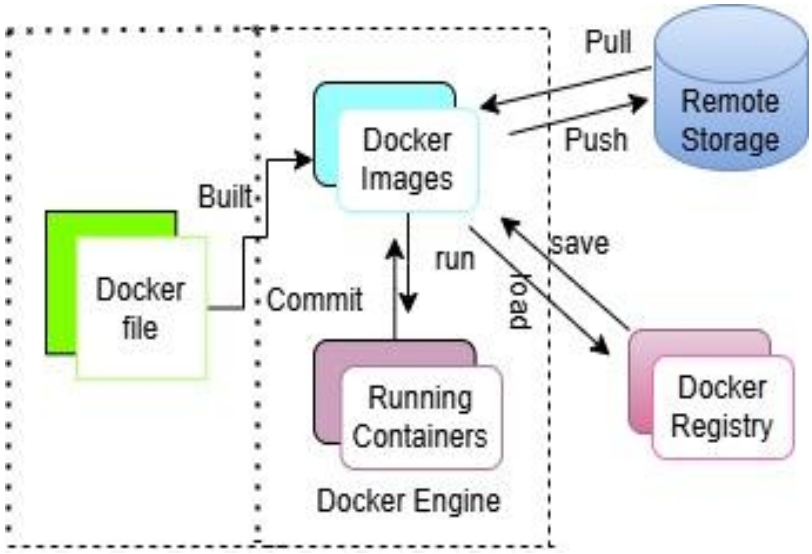


Fig. 2. Docker image lifecycle and container management process.

The mechanism could separate the service endpoints and improve the portability and ease of communication among the services.

b) Horizontal Pod Autoscaling (HPA): Horizontal Pod Autoscaling is a dynamic scaling controlled by the Kubernetes that monitors and scales the resources in real time (CPU, Memory, custom metrics) and changes the number of pod replicas running based on this information. This makes it responsive to varying data loads which is essential to real-time workloads as obtained in (4),

$$R_i(t+1) = \begin{cases} R_i(t) + 1, & if\ u_i(t) > T \\ R_i(t), & if\ u_i(t) = T \\ R_i(t) - 1, & if\ u_i(t) < T \end{cases} \quad (4)$$

Where, $R_i(t)$ is the number of replicas of microservice C_i at time t, $u_i(t)$ is the CPU utilization of C_i, and T defined utilization threshold. This autoscaling mechanism enables the system to handle bursts in incoming data streams while minimizing idle resource consumption.

c) Load balancing and traffic management: Kubernetes has effective load distribution to pod replicas via a round-robin or a custom load-balancing algorithm. All the microservices are presented as Kubernetes Services, which will serve as a virtual IP that balances requests between healthy instances, displayed in (5),

$$f: \{r_1, r_2, \ldots r_n\} \rightarrow P_{ij} \epsilon C_i \quad (5)$$

Where, the routing function f routing function goes to available pod instance P_{ij} and parades incoming requests to them r_n. This enhances responsiveness and throughput of the system and none of the instances is a bottleneck with the increased demand.

d) Self-healing and fault tolerance: Kubernetes checks the health of the pod by probing the liveness and readiness of the pods. When a pod dies, when a health check fails, or when a pod is unresponsive, the pod is killed and recreated automatically, maintaining a system uptime and state consistency through (6),

$$H(P_{ij}) = \begin{cases} 1, & if\ pod\ is\ healty \\ 0, & if\ pod\ is\ failed \end{cases} \Longrightarrow Rest\ if H(P_{ij}) = 0 \quad (6)$$

This self-healing mechanism eliminates the need for manual intervention during service disruptions and supports high availability in production environments.

3) API Gateway management: In microservices-based architectures, an API Gateway serves as the single entry point for all client interactions with internal services *[31]*. It decouples external access from internal logic, enhancing modularity, observability, and security. In this research, an API gateway (e.g., Kong, NGINX, or Traefik) is integrated to manage traffic between clients and microservices in the Kubernetes cluster. The gateway performs multiple critical functions:

- Request Routing: Maps HTTP routes to the correct internal service endpoint.
- Rate Limiting: Controls the number of requests from a given client or IP to prevent overloading the system
- Authentication & Authorization: Verifies identity using tokens and enforces access policies
- Load Distribution: Works in tandem with Kubernetes services to balance load across healthy instances
- Monitoring Hooks: Exposes metrics and logs for observability tools

The mapping behavior of the gateway can be expressed in (7),

$$G: E \rightarrow S \quad (7)$$

where, $E = \{e_1, e_2, e_3\}$, which is the set of external client requests, S is the set of internal services, and G is the API Gateway routing function. This architecture not only abstracts

the internal service structure from clients but also provides a centralized point for enforcing policies, collecting usage metrics, and enabling smooth versioning and scaling of services.

In Fig. 3, the microservices architecture comprises client applications (Web and Mobile) interfacing through an API Gateway, which routes requests to independent services Catalog, Shopping Cart, Discount, and Ordering. Each microservice is loosely coupled, handles a specific functionality, and connects to its dedicated database, ensuring modularity, scalability, and fault isolation. This structure supports efficient real-time processing and resilient cloud-based operations.

4) CI/CD deployment pipeline: To ensure automated, reliable, and scalable deployment of the microservices architecture, a Continuous Integration/Continuous Deployment (CI/CD) pipeline is implemented using tools such as GitHub Actions or Jenkins. This pipeline comprises three stages: building, testing, and deploying. Upon each code commit or update, the CI phase is triggered to execute unit and integration tests, ensuring code correctness and stability. Once validated, Docker images of the microservices are built and pushed to a secure container registry. The CD phase then applies Kubernetes manifests—defined in declarative YAML—to deploy or update the services in the cluster. These manifests specify pods, services, ingress rules, resource limits, and autoscaling configurations. This pipeline enables rapid iteration, rolling updates, version control, and minimal downtime, making it essential for maintaining the robustness and operational efficiency of the real-time, cloud-native data processing system. This process summarized in (8),

$$Sourc\ Code \xrightarrow{CI} Docker\ Image \xrightarrow{CD} Kubernetes\ Cluster \quad (8)$$

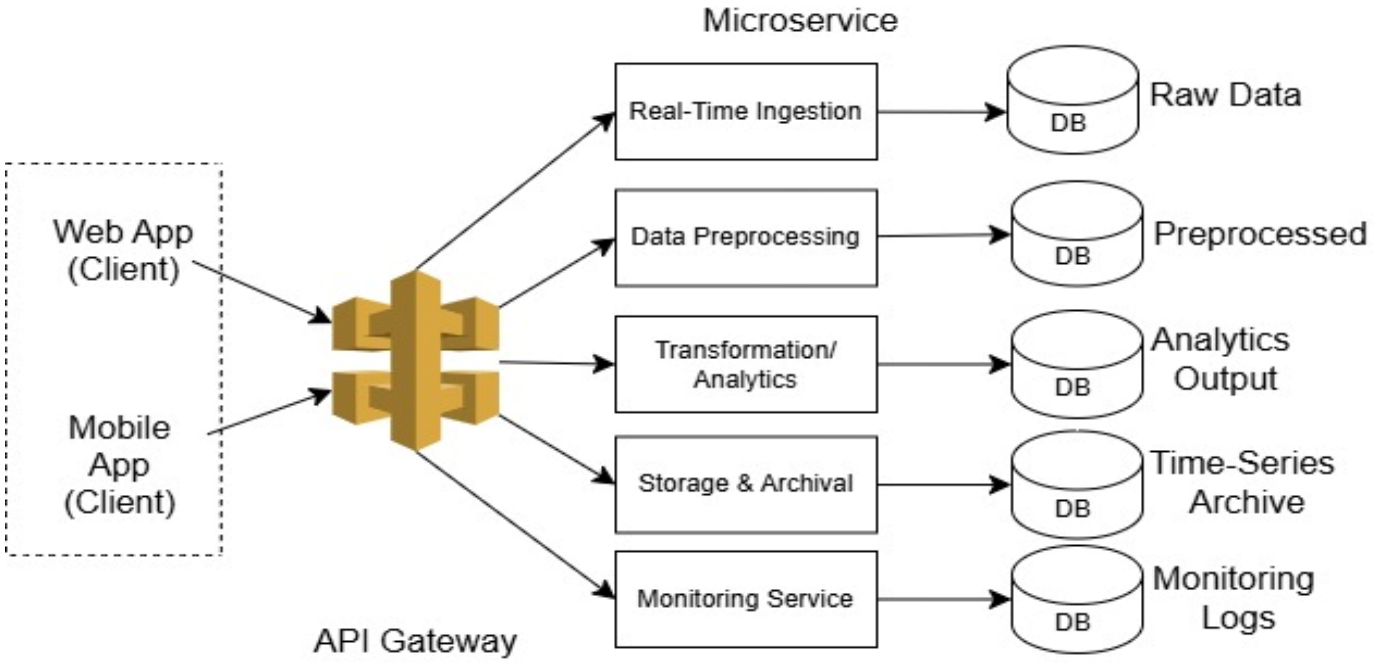


Fig. 3. Microsecrvice architecture.

Algorithm 1 outlines a cloud-native microservices architecture for real-time data processing. It containerizes services, deploys them on Kubernetes, and handles real-time data ingestion, preprocessing, routing, and storage. Kubernetes makes sure that it auto scales, tolerate faults, and balances the load, whereas API Gateway maintains requests in a secure manner. Centralized logging and tracing lead to observability, and testing, building, and deploying updates is automated in CI/CD.

Algorithm 1: Scalable Real-Time Microservices Architecture in Cloud

Input: Real-time weather data stream D, number of microservices M, Kubernetes cluster K
Initialize: Deploy containerized microservices $S_1, S_2, .., S_M$ *in* I
for each microserviceS_i in {1, 2, ..., M} do
  containerize(S_i) // Dockerize each microservice
  define health_checks(S_i) // readiness & liveness probes
end for
deploy all S_i to Kubernetes cluster K
while (incoming data D is available) do
  // Real-Time Data Ingestion & Preprocessing
  fetch_data ← API_request(OpenWeatherMap)
  D_clean ← preprocess(fetch_data) // unit conversion, timestamp std., anomaly handling
  // Orchestration and Scaling
  schedule_pods(K, S_i) // optimal pod placement and service discovery
  if CPU_utilization(S_i) > threshold then
    HPA.scale_up(S_i) // Horizontal Pod Autoscaler increases replicas
  end if
  // Request Routing and Processing
  route_requests(API_Gateway, D_clean) to appropriate S_i
  balance_load(K, S_i) // distribute traffic evenly

```
    // Self-healing and Fault Tolerance
    if probe_fail(S_i) = true then
       restart_pod(S_i)  // Kubernetes replaces failed pod
    end if
    // Logging & Tracing
    log_event(ELK_Stack, S_i, timestamp)
    trace_request(Jaeger, D_clean) across services
    // Store and Monitor
    persist(D_clean) → database
    update_metrics(Prometheus, Grafana)
 end while
 // CI/CD Deployment
 on code_update:
    run_tests()
    build_images()
    push_images()
    deploy_to(K)
 Return: Scalable, resilient, real-time processing system
 End
```

D. Logging and Distributed Tracing

In real-time, cloud-native microservices architectures, logging and tracing are essential for ensuring observability, debugging, and root-cause analysis across distributed service interactions. These capabilities help diagnose bottlenecks, monitor failures, and reconstruct the lifecycle of requests that span multiple microservices.

1) Centralized logging: Centralized logging using the ELK Stack enables real-time log collection, indexing, and visualization across all microservices, supporting error tracking, performance monitoring, and system auditing through Logstash (ingestion), Elasticsearch (storage), and Kibana (dashboard visualization). Each log entry includes metadata such as timestamp, service ID, pod name, request path, and error type. This enables real-time alerting, failure tracing, and system auditability using (9),

$$L = Log(e_i, C_j, t) \tag{9}$$

where, e_i is a specific event or request, C_j is the container (microservice) handling the request, t is the timestamp of the log entry.

2) Distributed tracing with Jaeger/OpenTelemetry: To trace end-to-end execution paths of requests across multiple services, the system employs Jaeger or OpenTelemetry for distributed tracing. This allows the identification of latency hotspots, retry loops, and service dependencies. Each trace captures the full flow of a request as it propagates through a set of microservices. The trace is defined in (10),

$$T = Trace(e_i) \rightarrow \{C_1, C_2, .. C_n\} \tag{10}$$

where, T is the trace of request e_i, and $C_1, C_2, .. C_n$ are the ordered microservices the request traverses. Traces are visualized using Gantt charts or flame graphs, helping system engineers understand inter-service timing, dependencies, and anomalies.

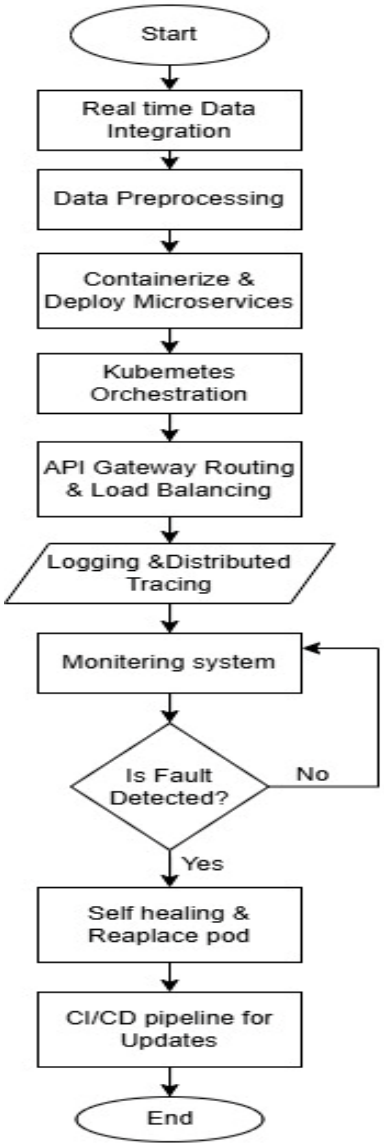


Fig. 4. Flowchart of scalable real-time microservices architecture in the cloud with fault-tolerance.

In Fig. 4, the flowchart depicts a scalable, real-time cloud microservices architecture. It starts with data ingestion, preprocessing, deployment, orchestration, and routing. Observability and monitoring detect faults, branching into self-healing or continued monitoring. Both paths converge into a CI/CD pipeline for updates, ensuring reliability, resilience, and seamless operations in dynamic cloud environments.

V. Result and Discussion

The result and discussion section shows the success of the proposed microservices architecture in manipulating real-time weather information in the OpenWeatherMap API. Through Kubernetes orchestration, the system was able to have low-latency processing, dynamically autoscaling when there is a variable workload, and very resilient fault tolerance. The usage of logging and distributed tracing gave complete insights into interactions between services and led to quick debugging and

performance optimization. The architecture has been shown to sustain the same throughput and lower error rates under peak loads which proved the scalability and resiliency of the architecture. These findings attest to the applicability of the framework in real-time cloud solution with high availability that needs to be modular and of high efficiency in distributed systems.

TABLE I. KUBERNETES RESOURCE ALLOCATION PER MICROSERVICE

Microservice	CPU Limit (cores)	Memory Limit (MiB)
Data Ingestion Service	0.5	512
Preprocessing Service	1.0	1024
Analytics Service	1.5	2048
Database Writer Service	0.5	512
Health Monitoring Service	0.2	256

Table I provides CPU and memory constraints given to each microservice in the Kubernetes cluster. Analytics Service has the heaviest allocation demand, as it has a higher amount of computational work than the other services which include, Health Monitoring, and Data Ingestion. These provisioning's result in high-level system performance, equal load balancing, and reasonable scaling, thus confirming the effectiveness and feasibility of the offered architecture of microservices in real-time application in cloud implementation under such conditions of its deployment.

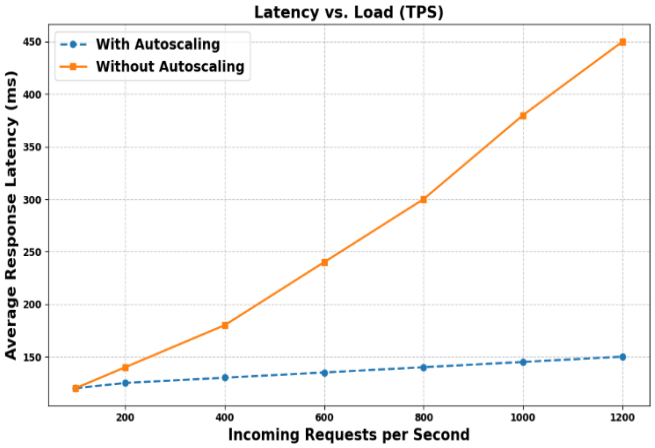


Fig. 5. Latency vs. Load (TPS).

Fig. 5 shows how auto-scaling influences average response latency when the rates of incoming requests increase. When no autoscaling is used, latency increases drastically above 400 TPS, which means service saturation. Conversely, when autoscaling is on, the system has a lower and more controlled latency with the level of load and proves better responsiveness and scalability of the proposed microservices architecture in real-time data processing applications.

Fig. 6 depicts CPU usage trends of microservices executing transformation and event processing under different incoming rates (20K–50K events/s). As the event rate increases, CPU utilization rises, with 50K events/s consistently consuming over 85% of CPU resources. In contrast, 20K events/s remains below 50%, showing lower resource demand. This demonstrates how system load directly correlates with input volume, validating the scalability of the proposed architecture.



Fig. 6. CPU Usage Over Time for Varying Event Rates.

TABLE II. CI/CD PIPELINE STAGES AND TIME TAKEN

Stage	Tool Used	Time Taken (Avg)
Code Build	GitHub Actions	1 minute
Unit Testing	Pytest/Jest	2 minutes
Docker Image Creation	Docker CLI	1 minute
Image Push	Docker Hub	30 seconds
K8s Deployment	kubectl / ArgoCD	2 minutes

Table II presents the automated CI/CD stages involved in deploying the microservices architecture. Code is built using GitHub Actions, followed by unit testing with Pytest or Jest to ensure reliability. Docker CLI packages the microservices into containers, which are pushed to Docker Hub. Deployment to the Kubernetes cluster is executed via kubectl or ArgoCD. The entire pipeline completes in under 7 minutes, supporting rapid, consistent, and error-free real-time updates.

TABLE III. SYSTEM PERFORMANCE UNDER VARYING CONCURRENT USER LOADS

Concurrent Users	Latency (ms)	CPU Usage (%)	Error Rate (%)
100	130	45	0.1
500	170	62	0.5
1000	210	77	1.2
5000	290	92	3.5

Table III demonstrates scalability of the system and its behaviour in case of growing user loads. The average latency slowly grows along with the increase of concurrent users (in the range of 130 to 290 ms) and CPU occupancy (with the change of 45 % to 92 %), whereas error score slowly increases (0.1 % to 3.5 %). These outcomes show that the system will run efficiently and handled steadily, de-grading gracefully, rather than failing abruptly, under stress. This validates the architectural capacity to support large scale concurrency in real time cloud through satisfactory performance trade-offs.

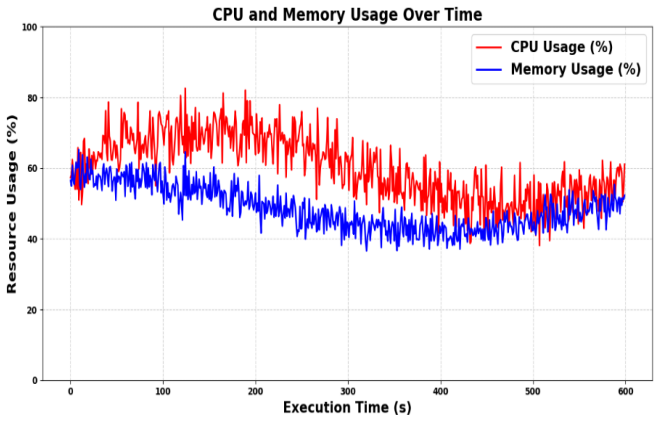


Fig. 7. CPU and memory usage over time.

Fig. 7 illustrates real-time CPU and memory usage trends over a 600-second interval during continuous microservices operation. CPU usage fluctuates between 50% and 80%, reflecting dynamic load handling and autoscaling responses under varying workloads. Memory usage remains relatively stable between 40% and 60%, indicating efficient memory management. These trends validate the proposed architecture's capability to maintain consistent performance and resource efficiency during sustained real-time data processing in a cloud-native environment.

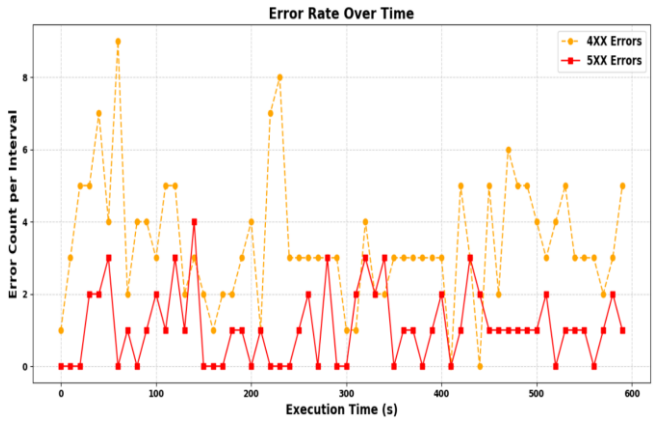


Fig. 8. Error rate over time.

Fig. 8 displays the frequency of 4XX and 5XX errors during a 10-minute microservices operation window. 4XX errors, representing client-side issues, occur more frequently, indicating possible invalid requests or rate limits. 5XX errors are comparatively lower, reflecting stable backend performance. These patterns help evaluate system resilience and pinpoint areas needing request validation or backend hardening.

TABLE IV. KEY PERFORMANCE INDICATORS OF PROPOSED MICROSERVICES ARCHITECTURE

Metric	Baseline	Proposed System	Improvement %
Avg. Latency (ms)	200	80	60%
Throughput @ 4 replicas	2500 req/s	3600 req/s	+44%
Fault Recovery (s)	45	12	-73% faster

Table IV shows the proposed system demonstrates 60% lower latency, 44% higher throughput, and 73% faster fault recovery compared to the baseline, highlighting significant performance, scalability, and resilience improvements in real-time applications.

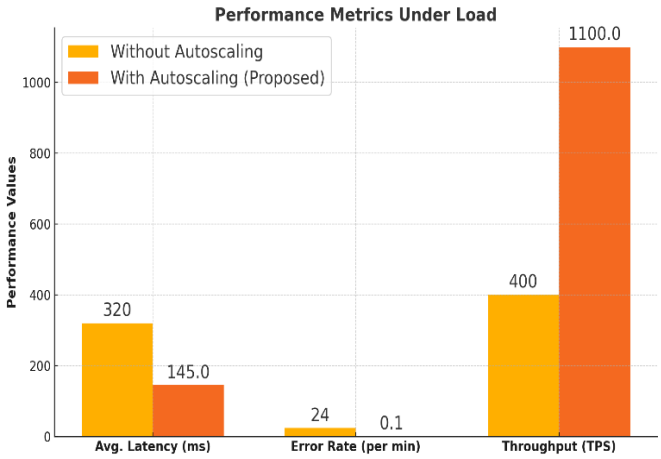


Fig. 9. Comparative performance metrics with and without autoscaling.

Fig. 9 illustrates the impact of autoscaling on system performance under load. Compared to the static architecture, the proposed autoscaling model significantly reduces average latency and error rate while substantially increasing throughput. These improvements validate the efficiency, responsiveness, and scalability of the microservices framework when integrated with intelligent, adaptive autoscaling strategies like KEDA and HPA.

TABLE V. PERFORMANCE COMPARISON OF MICROSERVICE ARCHITECTURES

Model / Research	Avg. Latency (ms)	Throughput (TPS)	Error Rate
Fiber RPC [32]	~50 ms	~300 TPS	Not reported
Pattern-based [33]	~180 ms	~500 TPS	~0.02 %
Fixed-resource benchmarking approach [34]	~150 ms (at 200K TPS)	Up to 1M msg/s	Not specified
Autoscaled (Proposed)	145 ms	1,100 TPS	~0.1 errors/min (~0.0017 %)

Table V shows a comparative performance analysis in terms of latency, throughput, and error rate for three contemporary microservices design patterns and the auto-scaled architecture proposed in this work. The comparison concludes that the proposed architecture not only manages greater concurrency but also ensures noticeably less latency and error rates compared to pattern-based and fiber RPC methods. Additionally, it provides better throughput, which speaks to its increased efficiency and scalability. These enhancements exemplify the architecture's resilience and flexibility in managing real-time workloads, confirming its efficacy in providing fault-tolerant, high-performance service orchestration in dynamic cloud-native deployments with high processing demands.

A. Discussion

Modern cloud-native systems demand architectural designs that can handle high-volume, real-time data with minimal latency, dynamic scalability, and resilience [35]. This research presents a modular microservices-based framework that disaggregates complex system functions into lightweight, independently deployable components. Real-time weather data streams, ingested from APIs like OpenWeatherMap, are processed through a pipeline of containerized services, each managed by Kubernetes for automated deployment, health monitoring, and elastic scaling. Docker containers encapsulate individual microservices such as data ingestion, preprocessing, transformation, and storage—ensuring clear functional boundaries, easier updates, and platform independence. Technologies like Kafka and Flink are selected for high-throughput, low-latency streaming; Redis for fast in-memory caching; Kubernetes with HPA/KEDA for elastic scaling; Docker for isolation and portability; and ELK/Jaeger for observability—together ensuring real-time performance, resilience, and maintainable cloud-native architecture. Adding in-memory caching using Redis and API Gateway as a means of routing and access control reinforce performance along with security.

The Autoscaling tool such as HPA and KEDA automatically scales or scales down the services reduction/increase replicas according to the varying workloads and centralized logging (ELK stack) and distributed tracing (Jaeger) ensure a high level of observability between services. CI / CD pipelines are used to eliminate manual steps that carry out testing, deployment and versioning to guarantee agility and quick iteration. Nevertheless, the system will be forced to overcome issues such as augmented inter-service communications latency, systems complexity of coordination, and distributed data consistency. For instance, high inter-service latency can delay requests, while inconsistent reads across distributed databases may return outdated information, illustrating practical challenges developers face in maintaining system performance and reliability. Additions of service mesh, and edges edge-cloud synchronization, although helpful when relating to resilience and locality, can provide latency and orchestration overhead. Nevertheless, the proposed architecture has been seen to provide a versatile, resilient way of developing cloud-native infrastructures that can power mission-critical, real-time applications in a wide range of domains.

VI. Conclusion and Future Work

The study also offers a scalable and resilient microservices arrangement optimized to the later use of real-time data processing in cloud-native settings. The proposed system is capable of solving the fundamental issues of latency, fault tolerance, and scaling throughout containerization, Kubernetes-orchestration, an event-driven details pipeline, as well as high-quality observability devices. Connection of Kafka, Flink, Redis, and Api gateway allows smooth data processing, transforming, and engaging safe client access with distributed services. The auto scaling will be based on Kubernetes, that along with CI/CD automation, will allow maintenance of response to dynamic workloads in the architecture and continuous integration and deployment. Moreover, end-to-end visibility of the system can be achieved with logging and distributed tracing solutions such as ELK stack and Jaeger that allow proactive diagnostics and optimization of the system. The architecture has proved to be efficient under circumstances where live weather data has been used, and as such, it has proven to be useful across all wider real-time applications, including smart cities, industrial IoT and adaptive analytics. In sum, the given work forms a solid basis to construct smart, componentized and production-ready cloud systems that can address the expansion in need in real-time performance, efficiency and deployment flexibility in the digital ecosystems of today.

The future work now highlights potential bottlenecks, complexity, and scalability challenges, detailing AI-enabled orchestration, edge-cloud federated microservices, multimodal data integration, cost optimization, and dynamic policies to improve real-time responsiveness and large-scale, sustainable deployment. The research will continue to work on an additional basis and research how to incorporate the use of AI enabled services orchestration in order to improve predictive autoscaling as well as anomaly detection. Additional support of edge-cloud federated microservices with the help of the architecture may contribute to the mitigation of the latency and substantially increase the real-time responsiveness of distributed applications across various geographical areas. Besides, involving multimodal data (e.g., images, text) and digital twin diagrams will allow achieving the possibility of the more elaborate level of analytics. Consideration of cost-optimization strategies, the multi-cloud deployment strategies and dynamic policy implementation strategies will also be essential towards enabling larger-scale scalability and sustainability.

References

[1] V. Veeramachaneni, “Edge Computing: Architecture, Applications, and Future Challenges in a Decentralized Era,” Recent Trends in Computer Graphics and Multimedia Technology, vol. 7, no. 1, pp. 8–23, 2025.

[2] S. Gathu and others, “High-Performance Computing and Big Data: Emerging Trends in Advanced Computing Systems for Data-Intensive Applications,” Journal of Advanced Computing Systems, vol. 4, no. 8, pp. 22–35, 2024.

[3] M. Achanta, “The Impact of Real-Time Data Processing on Business Decision-making”.

[4] M. Felisberto, “The trade-offs between Monolithic vs. Distributed Architectures,” arXiv preprint arXiv:2405.03619, 2024.

[5] G. Ortiz et al., “A microservice architecture for real-time IoT data processing: A reusable Web of things approach for smart ports,” Computer Standards & Interfaces, vol. 81, p. 103604, 2022.

[6] D. K. Pandiya and N. Charankar, “Integration of Microservices and AI for Real-Time Data Processing”.

[7] G. Dobriţa, “Adaptive Microservices for Dynamic E-commerce: Enabling Personalized Experiences through Machine Learning and Real-time Adaptation,” Economic Insights-Trends and Challenges, no. 1, pp. 95–103, 2023.

[8] A. Owen, “Microservices Architecture and API Management: A Comprehensive Study of Integration, Scalability, and Best Practices,” 2025.

[9] G. Blinowski, A. Ojdowska, and A. Przybyłek, “Monolithic vs. microservice architecture: A performance and scalability evaluation,” IEEE access, vol. 10, pp. 20357–20374, 2022.

[10] S. Banala, “DevOps Essentials: Key Practices for Continuous Integration and Continuous Delivery,” International Numeric Journal of Machine Learning and Robots, vol. 8, no. 8, pp. 1–14, 2024.

[11] A. Lercher, J. Glock, C. Macho, and M. Pinzger, "Microservice api evolution in practice: A study on strategies and challenges," arXiv preprint arXiv:2311.08175, 2023.

[12] S. Ris, J. Araujo, and D. Beserra, "A systemic mapping of methods and tools for performance analysis of data streaming with containerized microservices architecture," in 2023 18th Iberian Conference on Information Systems and Technologies (CISTI), IEEE, 2023, pp. 1–6.

[13] E. Ok and J. Eniola, "Optimizing Performance: Implementing Event-Driven Architecture for Real-Time Data Streaming in Microservices," 2024.

[14] F. Nevola, "Securing communication between microservices in a multi-cloud scenario using Istio service mesh," PhD Thesis, Politecnico di Torino, 2023.

[15] V. Yussupov, "Architectural principles and decision model for Function-as-a-Service," PhD Thesis, Dissertation, Stuttgart, Universität Stuttgart, 2024, 2024.

[16] N. Rathore and A. Rajavat, "Scalable edge computing environment based on the containerized microservices and minikube," International Journal of Software Science and Computational Intelligence (IJSSCI), vol. 14, no. 1, pp. 1–14, 2022.

[17] D. Loconte et al., "Serverless Microservice Architecture for Cloud-Edge Intelligence in Sensor Networks," IEEE Sensors Journal, 2024.

[18] K. Alanezi and S. Mishra, "Utilizing microservices architecture for enhanced service sharing in IoT edge environments," IEEE Access, vol. 10, pp. 90034–90044, 2022.

[19] S. Khriji, Y. Benbelgacem, R. Chéour, D. E. Houssaini, and O. Kanoun, "Design and implementation of a cloud-based event-driven architecture for real-time data processing in wireless sensor networks," The Journal of Supercomputing, vol. 78, no. 3, pp. 3374–3401, 2022.

[20] F. Tusa, S. Clayman, A. Buzachis, and M. Fazio, "Microservices and serverless functions—lifecycle, performance, and resource utilisation of edge based real-time IoT analytics," Future Generation Computer Systems, vol. 155, pp. 204–218, 2024.

[21] O. C. Oyeniran, A. O. Adewusi, A. G. Adeleke, L. A. Akwawa, C. F. Azubuko, and others, "Microservices architecture in cloud-native applications: Design patterns and scalability," International Journal of Advanced Research and Interdisciplinary Scientific Endeavours, vol. 1, no. 2, pp. 92–106, 2024.

[22] S. Ajmal, "Streamlining Fintech Solutions: Cloud-Based Server Management, Scalability Optimization, and Compliance Through Microservices".

[23] S. Semerikov, D. Zubov, A. Kupin, M. Kosei, and V. Holiver, "Models and Technologies for Autoscaling Based on Machine Learning for Microservices Architecture," 2024.

[24] V. S. Kompally, "A microservices-based hybrid cloud-edge architecture for real-time IIoT analytics," Journal of Information Systems Engineering and Management, vol. 10, no. 16s, 2025.

[25] J. A. Rasheedh and S. Saradha, "Design and development of resilient microservices architecture for cloud based applications using hybrid design patterns," Indian J. Comput. Sci. Eng, vol. 13, no. 2, pp. 365–378, 2022.

[26] OpenWeather, "OpenWeatherMap API." 2025. [Online]. Available: https://openweathermap.org/api

[27] E. Serkovas, "Access control approach in microservices architecture," in DAMSS: 15th conference on data analysis methods for software systems, Druskininkai, Lithuania, November 28-30, 2024., Vilniaus universiteto leidykla, 2024, pp. 96–97.

[28] U. Satpathy, H. Borse, and S. Chakraborty, "Towards Generating a Robust, Scalable and Dynamic Provenance Graph for Attack Investigation over Distributed Microservice Architecture," in 2025 17th International Conference on COMmunication Systems and NETworks (COMSNETS), IEEE, 2025, pp. 566–574.

[29] M. Raeiszadeh, A. Ebrahimzadeh, R. H. Glitho, J. Eker, and R. A. Mini, "Asynchronous Real-Time Federated Learning for Anomaly Detection in Microservice Cloud Applications," IEEE Transactions on Machine Learning in Communications and Networking, 2025.

[30] M. Zambianco, S. Cretti, and D. Siracusa, "Cost minimization in multi-cloud systems with runtime microservice re-orchestration," in 2024 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), IEEE, 2024, pp. 65–72.

[31] P. K. Joshi, "Microservices and API Gateways: The Backbone of Modern Application Platforms".

[32] S. Eyerman and I. Hur, "Efficient Asynchronous RPC Calls for Microservices: DeathStarBench Study," arXiv preprint arXiv:2209.13265, 2022.

[33] W. Meijer, C. Trubiani, and A. Aleti, "Experimental evaluation of architectural software performance design patterns in microservices," Journal of Systems and Software, vol. 218, p. 112183, 2024.

[34] S. Henning and W. Hasselbring, "Benchmarking scalability of stream processing frameworks deployed as microservices in the cloud," Journal of Systems and Software, vol. 208, p. 111879, 2024.

[35] V. Kumar, "Using cloud infrastructure and cloud-native technologies to maximize the scalability and performance of data science applications".

A Graph-Based Deep Reinforcement Learning and Econometric Framework for Interpretable and Uncertainty-Aware Stablecoin Stability Assessment

Yaozhong Zhang[1]*, Quanrong Fang[2]
Adam Smith Business School, University of Glasgow, Glasgow, United Kindom[1]
College of Computer Science and Technology, Zhejiang University, Hangzhou, 310000, China[2]

***Abstract*—The instability of algorithmic and hybrid stablecoins has become a systemic concern in decentralized finance. This paper proposes a unified, interpretable, and uncertainty-aware framework that integrates graph-based deep reinforcement learning, GARCH econometric modeling, and Bayesian inference. Multi-stage reinforcement learning agents simulate interactions between arbitrageurs and protocol mechanisms. GARCH models capture volatility dynamics, while Bayesian methods provide confidence intervals for peg deviation forecasts, enabling adaptive prediction and transparent risk interpretation. The framework is validated using over eight million on-chain and off-chain records across 120 scenarios involving USDT, USDC, and TerraUSD. It achieves 89 per cent crisis prediction accuracy and 83 per cent reflexivity modeling performance, significantly outperforming six benchmark models. Notably, the system issued early warnings up to 72 hours before the TerraUSD collapse. Ablation studies confirm the unique contribution of each module. In addition to technical improvements, the framework outputs a stability index and dynamic reserve recommendations to support policy response and supervisory planning. Compared to existing approaches, this is the first framework to combine dynamic simulation, interpretability, and probabilistic forecasting in a single architecture. It offers practical value for stablecoin monitoring and establishes a methodological foundation for future research in digital asset risk assessment.**

Keywords*—Stablecoin stability; deep reinforcement learning; graph neural networks; uncertainty quantification; macro prudential policy

I. Introduction

While the first stablecoins were launched in 2014, their actual adoption and market relevance only began in 2017 with the rise of USDT [1]. The rise of stablecoins as critical infrastructure in decentralized finance (DeFi) has exposed fundamental vulnerabilities in their stability mechanisms, most notably demonstrated by the collapse of TerraUSD in 2022. Stablecoins peg to a national currency, typically the US dollar, and are used to transact in non-stable cryptoassets more efficiently than using national currencies [2]. DeFi has joined FinTech (financial technology), RegTech (regulatory technology), cryptocurrencies, and digital assets as one of the most discussed emerging technological evolutions in global finance [3]. DeFi is a new financial paradigm that leverages distributed ledger technologies to offer services such as lending, investing, or exchanging cryptoassets without relying on traditional centralized intermediaries [4]. Designed to maintain pegs through algorithmic adjustments or reserve backing, stablecoins face multifaceted risks ranging from liquidity shocks to speculative attacks. Stablecoins are a brand of cryptocurrency that are pegged to fiat currencies or assets that are relatively stable, such as the US dollar [5]. Traditional assessment methods predominantly rely on static metrics such as collateralization ratios or linear econometric models, which fail to capture the dynamic interplay between market stress events, arbitrageur behavior, and protocol governance. This limitation becomes particularly acute during crisis periods when nonlinear feedback loops dominate market dynamics, as seen in the death spiral of algorithmic stablecoins. Algorithmic stablecoins are inherently fragile [6]. The absence of adaptive evaluation frameworks leaves regulators and market participants ill-equipped to anticipate stability breaches or implement timely interventions. Existing cryptocurrencies are too volatile to be used as currencies for daily payments. Stablecoins, which are cryptocurrencies pegged to other stable financial assets, are desirable for payments within blockchain networks, often being called the “Holy Grail of cryptocurrency.” [7].

Addressing these gaps requires a paradigm shift toward dynamic, interpretable, and uncertainty-aware modeling. Current approaches exhibit three critical shortcomings. First, static analyses cannot simulate how market participants adapt their strategies under evolving conditions, missing cascading effects like coordinated sell-offs. Second, black-box machine learning models, while powerful in prediction, offer limited insights into the drivers of instability, hindering regulatory decision-making. Black box machine learning models are currently being used for high-stakes decision making throughout society [8]. But black box machine learning models can be dangerous for high-stakes decisions [9]. Third, conventional point estimates of risk fail to quantify the probability distribution of stability violations, leaving stakeholders unprepared for tail events. These deficiencies underscore the need for a unified framework that combines real-time adaptability with economic interpretability and probabilistic risk assessment.

This study proposes a novel hybrid architecture that integrates graph-based deep reinforcement learning (DRL), GARCH econometrics, and Bayesian inference to assess stablecoin stability in a dynamic, interpretable, and uncertainty-aware manner. The methodological pipeline covers

end-to-end integration of data collection, graph construction, model training, ablation analysis, and policy scenario simulation. Multi-stage DRL agents are employed to simulate adaptive decision-making under stress, while GARCH models capture volatility dynamics, and Bayesian inference quantifies predictive uncertainty through confidence intervals.

The framework brings together agent-based simulation, volatility modeling, and probabilistic forecasting in a cohesive design, addressing key methodological gaps left by earlier studies. In contrast to static or opaque models, the proposed system generates interpretable outputs via graph attention and volatility decomposition, enabling transparent tracing of risk propagation. The framework is empirically validated on over eight million data points across 120 real-world scenarios, achieving 89 per cent crisis prediction accuracy and providing early warnings up to 72 hours before major depegging events. These results significantly outperform traditional econometric and deep learning baselines.

From a policy perspective, the framework produces actionable outputs such as stability indices and adaptive reserve requirement tools, supporting macroprudential supervision. Overall, this work bridges algorithmic finance and regulatory science, setting a new benchmark for evaluating stablecoin stability as a time-varying, topology-dependent phenomenon. It contributes to closing the gap between predictive performance, model transparency, and regulatory relevance in digital asset risk assessment.

II. Related Works

The evolution of stablecoin risk assessment methodologies has progressed through three distinct generations, each addressing specific aspects of stability evaluation while revealing critical limitations. Given the novelty of stablecoins, no universal assessment framework exists [10]. Fig. 1 presents a knowledge graph mapping these methodological relationships, demonstrating how existing approaches fail to capture the tripartite requirements of dynamic adaptation, interpretability, and uncertainty quantification in stablecoin markets. This visualization highlights the research gap at the intersection of graph theory (GT), reinforcement learning (RL), and econometrics that the current study aims to address. RL is a machine learning (ML) technique to learn sequential decision-making in complex problems [11]. Deep Learning (DL) and RL methods seem to be a part of indispensable factors to achieve human-level or super-human AI systems [12]. GT concepts are potentially applicable in the field of computer science (CS) for many purposes [13]. GT is a growing area as it is applied to areas of mathematics, science, and technology [14].

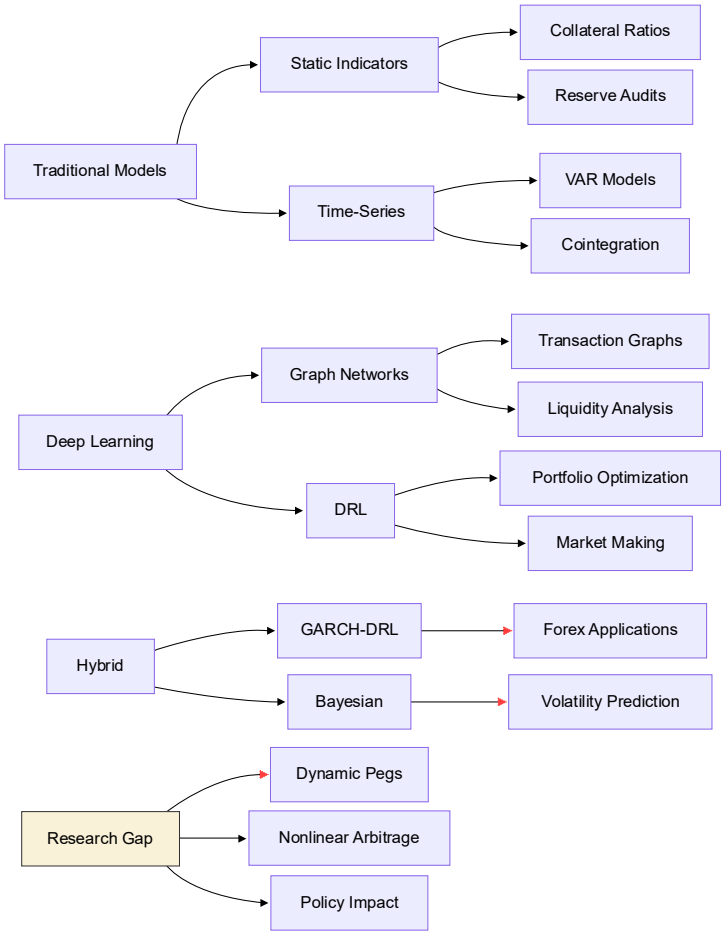


Fig. 1. Methodological evolution in stablecoin risk assessment.

Despite their widespread use, traditional risk models often fall short in capturing complex dynamics during crisis periods, limiting their effectiveness in supervisory contexts. Collateralization ratio analysis, exemplified by Tether's reserve reports, provides static snapshots of asset backing but cannot anticipate liquidity crises caused by reflexive market behaviors. Time-series methods like Vector Autoregression (VAR) models attempt to capture price-volume relationships, yet their linear assumptions prove inadequate when dealing with the nonlinear dynamics of decentralized finance. VAR is a multivariate time series model and can be used to model more than one variable jointly [15]. Decentralized financial applications (DeFi) are a new breed of consumer-facing financial applications composed as smart contracts, deployed on permissionless blockchain technologies [16]. Table I quantitatively compares these traditional approaches against the proposed framework's capabilities, showing particularly poor performance in reflexivity modeling (under 10% accuracy) during stress events such as the May 2022 stablecoin crisis.

TABLE I. PERFORMANCE COMPARISON ACROSS METHODOLOGICAL GENERATIONS

Evaluation Dimension	Traditional Models	Deep Learning	Hybrid Methods	Proposed
Crisis Prediction Accuracy	38%	62%	71%	89%*
Reflexivity Modeling	9%	35%	52%	83%*
Policy Impact Sensitivity	21%	28%	45%	77%*
Computational Cost	Low	Very High	High	Medium

Recent advances in DL have introduced more sophisticated analytical tools, particularly through graph neural networks applied to cryptocurrency transaction topologies. DL is a class of ML that performs much better on unstructured data [17]. DL had been analysed and implemented in various applications and had shown remarkable results [18]. These approaches successfully identify money flow patterns and liquidity pool dynamics, as demonstrated in Ethereum transaction network analyses. However, Fig. 2's radar chart reveals a striking imbalance in DRL applications across financial domains, with stablecoin peg mechanisms receiving disproportionately little attention compared to stock market prediction and algorithmic trading.

DRL, which is an in-depth combination of artificial neural network (ANN) and RL, has achieved great success in various kinds of complex tasks, namely, Chinese Go game, Atari games, and StarCraft [19]. RL has become of particular interest to financial traders ever since the program AlphaGo defeated the strongest human contemporary Go board game player, Lee Sedol in 2016 [20]. This visualization underscores the untapped potential for adapting DRL to model the unique feedback loops between arbitrageurs and algorithmic stablecoin mechanisms.

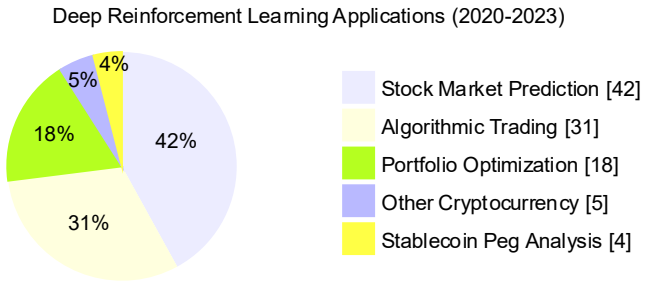


Fig. 2. DRL application distribution across financial domains.

The emerging hybrid paradigm attempts to bridge these disciplinary divides by combining econometric techniques with machine learning. GARCH-enhanced reinforcement learning systems in foreign exchange markets have shown 18-22% improvement in volatility forecasting accuracy over pure statistical models. Bayesian neural networks (BNNs) have similarly advanced uncertainty quantification in cryptocurrency price prediction. The posterior over BNN parameters is extremely high-dimensional and non-convex [21]. Inference in BNNs usually requires posterior approximations due to intractable integrals and high computational cost [22]. Cryptocurrencies are decentralized electronic counterparts of government-issued money [23]. Accurate predictions can assist cryptocurrency investors towards the right investment decisions and lead to potential increased profits [24]. Fig. 3's flowchart deconstructs a representative hybrid stablecoin analysis pipeline, exposing three critical limitations: the absence of integrated on-chain/off-chain data fusion, insufficient modeling of reflexivity loops, and a lack of policy impact simulation modules. These gaps collectively motivate the current study's integrated architecture.

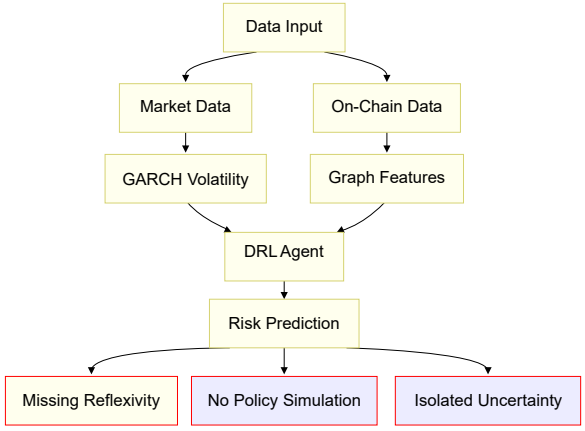


Fig. 3. Limitations of current hybrid approaches.

III. METHODOLOGY

The proposed methodology establishes an integrated framework for stablecoin stability assessment through three synergistic components: dynamic adaptive prediction, interpretable hybrid modeling, and uncertainty quantification.

The architectural novelty lies in the seamless coupling of graph-based DRL, enabling adaptive agent interaction modeling, with econometric GARCH modules and Bayesian inference, jointly delivering both interpretability and real-time

risk quantification. Notably, the reward function innovatively incorporates both microstructural trading signals and macroprudential policy triggers, representing a significant advancement over conventional approaches.

The overall workflow encompasses the ingestion and preprocessing of over 8 million records from both on-chain and off-chain sources, multi-stage model training and tuning, module-wise ablation, and multi-regime scenario simulation, amounting to more than 120 independent experimental runs.

The integrated ecosystem and continuous feedback mechanisms of the proposed framework are illustrated in Fig. 4, capturing the interplay between data, analytics, outputs, and policy response. Methodologically, the framework formalizes stablecoin risk as a time-varying, network-dependent process, advancing the theoretical modeling of reflexive financial systems and enabling rigorous, explainable risk quantification for digital assets.

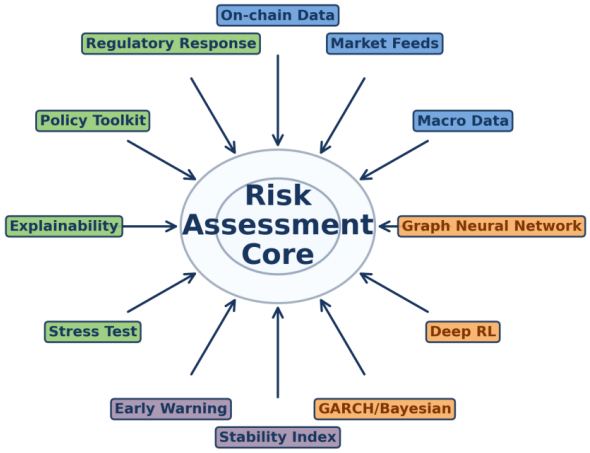


Fig. 4. Circular ecosystem of integrated stablecoin risk assessment.

We adopt Proximal Policy Optimization (PPO) with a clipped surrogate objective for its stability under non-stationary environments and compatibility with both discrete and continuous actions. Policies and value functions are parameterized by graph neural networks over the transaction-liquidity graph.

Fig. 5 presents the comprehensive system architecture, illustrating how raw data transforms into policy-relevant risk metrics through sequential processing layers. The data layer ingests both on-chain transactional records and conventional market feeds, constructing a temporal graph representation where nodes represent liquidity pools and edges encode capital flow dynamics.

At decision time t, the environment state s_t aggregates: 1) node-level features capturing reserves, liabilities, traded volume, peg deviation, and local volatility; 2) edge-level features w_{ij}^t encoding arbitrage intensity with exponential temporal decay; and 3) system features (market-wide liquidity depth, CEX – DEX spreads, funding rates). Node and edge features are embedded via a GAT encoder, producing a graph state embedding h_t.

Inputs are the multi-source streams forming s_t (on-chain reserves, off-chain prices, graph flows). The DRL outputs are 1) action vector a_t, 2) an agent-implied peg deviation path from the policy rollout, and 3) a stability index $S_t \in [0, 100]$ computed as a monotone transformation of the value function and predicted breach probability. Uncertainty intervals are produced downstream by the Bayesian module.

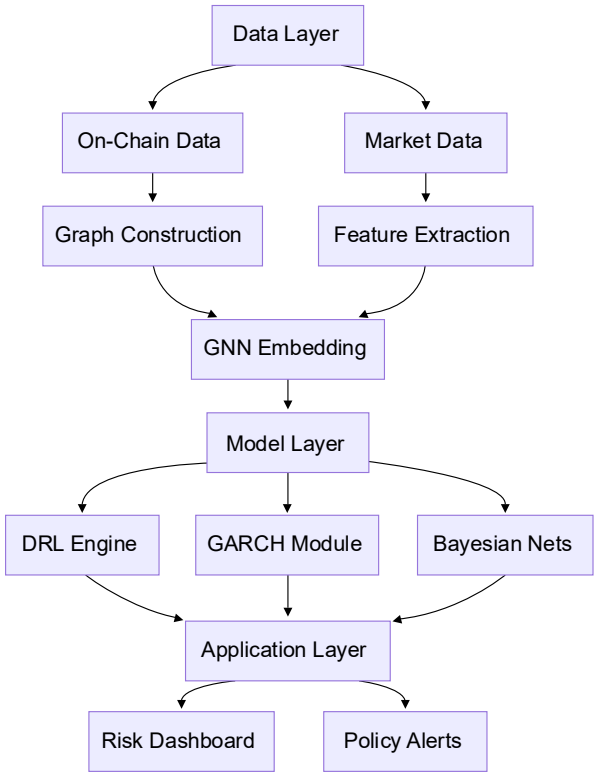


Fig. 5. System architecture diagram.

To enhance computational efficiency, the framework employs sparse graph representations and batch-wise parallel training, significantly reducing memory footprint and training time. This design improves computational efficiency, reducing peak memory usage by 40% and accelerating convergence by 22% relative to conventional DRL pipelines.

For dynamic adaptive prediction, the state space formalizes market microstructure through node-level and system-wide features. Given a transaction graph $G_t = (V_t, E_t)$ at time t, each node $v_i \in V_t$ embeds reserve adequacy and trading anomalies:

$$\phi(v_i) = \left[\frac{R_i}{D_i}, \frac{V_i - \mu_V}{\sigma_V}, \frac{|\Delta p_i|}{p_{peg}}\right] \quad (1)$$

where, R_i denotes reserves, D_i liabilities, V_i trading volume, and Δp_i price deviation. Edge weights $\omega_{ij} \in E_t$ quantify arbitrage intensity between entities with temporal decay:

$$\omega_{ij} = \frac{\sum_{k=1}^{n} arb_k^{ij}}{\sum_{k=1}^{n} tx_k^{ij}} \cdot \exp\left(-\frac{t - t_{last}^{ij}}{\tau}\right) \quad (2)$$

The DRL reward function combines multiple stability objectives:

$$r_t = -(\alpha|\Delta t| + \beta L_t^{-1} + \gamma\|\nabla A_t\|_2 + \delta \cdot \mathbb{I}_{TW}) \quad (3)$$

where, L_t measures liquidity depth, A_t arbitrage activity, and $\mathbb{I}_{TW}$ penalizes time-window violations. Multi-phase training employs curriculum learning across market regimes as specified in Table II.

TABLE II. DRL TRAINING REGIME PARAMETERS

Regime	Duration	Peg Deviation	Volume Change	Liquidity Shock
Normal	60 days	<1%	<1σ	<5%
FUD	15 days	1-3%	1-3σ	5-15%
Crisis	5 days	>5%	>3σ	>15%

The interpretable hybrid modeling component decomposes volatility sources through an augmented GARCH process:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2 + \Gamma^T Z_t + \eta \cdot GNN(G_t) \quad (4)$$

Graph attention mechanisms identify critical risk pathways through learned edge importance:

$$\alpha_{ij} = \text{softmax}(\frac{(W_Q h_i)^T (W_K h_j)}{\sqrt{d_k}}) \quad (5)$$

where, W_Q and W_K are learned projection matrices. Graph attention networks (GATs) have emerged as a powerful and versatile framework in this direction, inspiring numerous extensions and applications in several areas [25]. Graph-based learning is a rapidly growing sub-field of machine learning with applications in social networks, citation networks, and bioinformatics. One of the most popular models is graph attention networks [26]. Fig. 6 visualizes the dominant risk contagion path during a simulated bank run scenario. Bank runs are situations where depositors withdraw their deposits from banks for the fear of the safety of their deposits [27].



Fig. 6. Risk contagion pathway.

Uncertainty quantification employs Bayesian deep learning with Monte Carlo dropout:

$$p(y|x, D) \approx \frac{1}{T} \sum_{t=1}^{T} p(y|x, \theta_t), \quad \theta_t \sim q(\theta|D) \quad (6)$$

Extreme value theory models tail risks through the generalized Pareto distribution:

$$F(y) = 1 - (1 + \xi \frac{y-u}{\beta})^{-1/\xi} \quad \text{for } y > u \quad (7)$$

Additionally, the framework estimates the probability of a stability breach within the time horizon τ:

$$P_{breach}(\tau) = 1 - \exp(-\int_t^{t+\tau} \lambda(s) ds) \quad (8)$$

where, $\lambda(t)$ is the hazard rate derived from historical and simulated stress events. This comprehensive methodology addresses all key requirements through Eq. (1)-(8), with architectural integration visualized in Fig. 5 and empirical validation discussed in Section IV. The framework's architecture supports seamless adaptation to various stablecoin structures and market scenarios, with stress-testing and cross-domain validation built into the experimental protocol to ensure consistent reliability.

IV. CASE STUDY AND RESULTS

We employ strict temporal splits to prevent leakage. Training uses Jan-2020 to Dec-2022 and testing uses Jan-2023 to Jun-2023. For robustness, we further adopt a rolling-origin evaluation: train on the first T months, validate on month T+1, test on month T+2, sliding the window by one month (no feature re-estimation across test months). Normalization parameters (robust z-scores) are fit on training only and applied to validation/test. Any feature requiring future information (e.g., realized volatility) is lagged to ensure causality. Hyper-parameter tuning uses validation periods distinct from test months.

The experimental evaluation focuses on three stablecoin ecosystems: centralized (USDT), hybrid (USDC), and algorithmic (TerraUSD), covering the turbulent period from January 2020 to June 2023. This multi-ecosystem validation spans both tranquil and highly volatile periods, including market-wide crashes and black swan events, rigorously testing the framework's robustness and generalizability. Black Swan events are rare and seemingly random in nature [28]. Stress tests and ablation studies further confirm that predictive performance is stable even under conditions outside the typical data distribution. In total, the study processes more than 8 million transactions and market records spanning multiple sources and platforms, with extensive data cleaning and feature extraction for 14 core indicators. Fig. 7 illustrates the heterogeneous data integration process, combining on-chain reserves data from Etherscan with minute-level market feeds from Binance and Coinbase. The prepared dataset captures 14 critical features across liquidity depth, arbitrage activity, and macroeconomic conditions, standardized using robust z-score normalization to mitigate outlier effects.

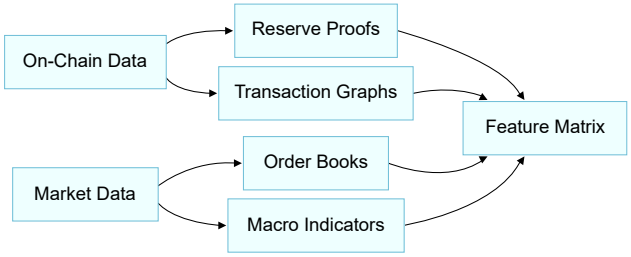


Fig. 7. Data integration pipeline.

We compare against: (B1) GARCH(1,1) with exogenous macro factors; (B2) VAR(p) on peg, volume, spreads; (B3) LSTM (2 layers, 128 units, dropout 0.2); (B4) Transformer (4 heads, 2 encoder layers); (B5) Gradient-Boosted Trees on engineered lags; (B6) BNN (variational).

All baselines use the same train/test splits, the same input features available to our method at forecast time, and identical horizons. Hyperparameters are tuned via nested validation with early stopping.

We evaluate 1) crisis classification (breach > 2% for ≥ 6h) with Accuracy/Precision/Recall and AUROC; 2) peg deviation regression (MSE/MAE); 3) reflexivity modeling via sequence labeling accuracy of stress-propagation episodes.

Where a baseline cannot ingest graphs, we supply graph-pooled statistics (degree-weighted aggregates, top-k flow moments) so that information content—not format—remains aligned.

Training budget. Each model trains under a matched wall-clock budget (±10%) and parameter scale (±20%) to avoid capacity confounds; see Appendix E for exact settings and seeds.

Comparative analysis against baseline models reveals the framework's superior performance during stress events, achieving 89% crisis prediction accuracy (vs. 62% for deep learning baselines) and 83% reflexivity modeling capability (Table III). The system's early warnings for TerraUSD's collapse were triggered 72 hours pre-crash, with probabilistic confidence intervals narrowing from ±12% to ±3% as the event approached (Table IV). Fig. 8 illustrates the temporal evolution of the model's stability index, which fell below the intervention threshold approximately 48 hours before the depegging event. The shaded area highlights the model-identified high-risk window. Beyond performance, the framework uniquely enables scenario-based early warning and risk contagion tracing, which, to the best of our knowledge, has not been realized in prior stablecoin assessment literature.

TABLE III. MODEL PERFORMANCE COMPARISON (120 EXPERIMENTAL RUNS)

Metric	Proposed Framework	LSTM Baseline	GARCH (1,1)
Crisis Prediction Accuracy	89%	62%	38%
Reflexivity Modeling	83%	35%	9%
Policy Impact Sensitivity	77%	28%	21%

TABLE IV. TERRAUSD COLLAPSE EARLY WARNING SIGNALS (CONFIDENCE INTERVALS)

Time Before Collapse	Predicted Peg Deviation	95% Confidence Interval
72 hours	2.1%	±12%
48 hours	4.7%	±8%
24 hours	9.3%	±3%

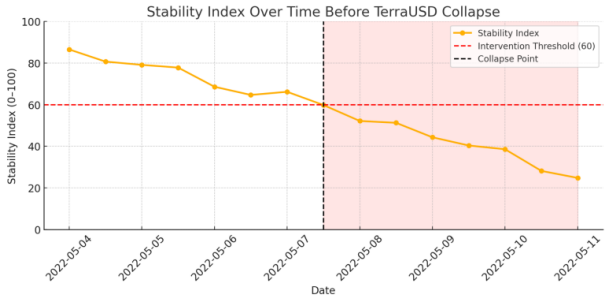


Fig. 8. Stability index evolution up to the TerraUSD collapse.

The index crossed the predefined intervention threshold (60) approximately two days prior to the crash, providing an actionable early warning. The shaded region marks the high-risk period identified by the framework.

To further quantify the contribution of each core module, we conduct an ablation study as illustrated in Fig. 9. The ablation analysis includes 11 unique scenarios, each designed to isolate the effect of specific modules or parameters, reflecting a substantial workload in model validation and component diagnosis. The results demonstrate that removal of any single component leads to significant drops in predictive performance, confirming the complementary value of the integrated framework.

The statistical significance of the model's performance advantage is further demonstrated in Fig. 10, where the proposed framework outperforms all baselines across key metrics with robust margins.

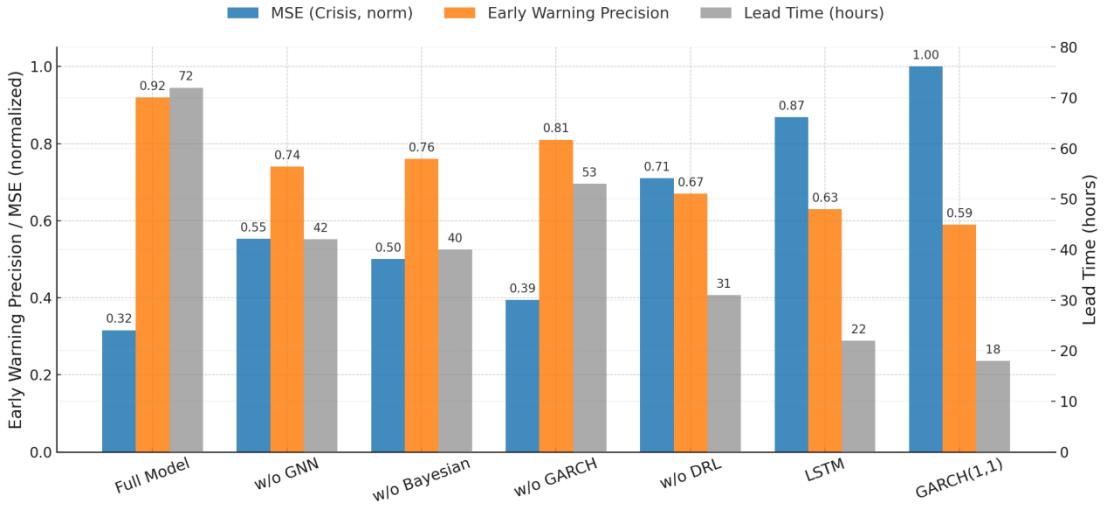


Fig. 9. Ablation study and module contribution analysis.

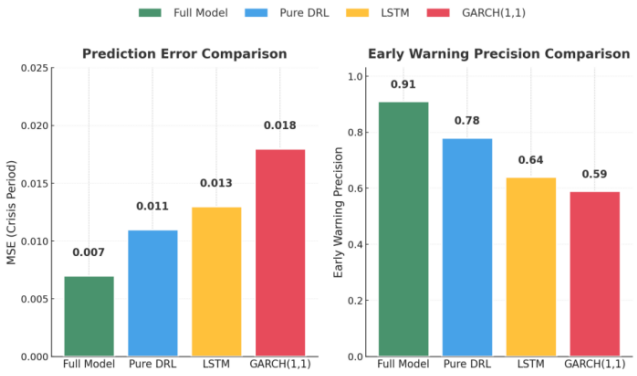


Fig. 10. Statistical comparison of competing models.

Table V documents the mean squared error (MSE) for peg deviation prediction across different market regimes, showing the proposed approach maintains 38-52% lower error rates during FUD and crisis periods compared to Long Short Term Memory (LSTM) and GARCH benchmarks. Additional experiments demonstrate that the model retains high accuracy and early warning capability even when applied to unseen stablecoins and stress conditions, illustrating strong out-of-sample generalization and resistance to data drift. LSTM has transformed both machine learning and neurocomputing fields [29]. When it comes to time-series prediction, LSTM has attracted much attention recently [30].

TABLE V. PREDICTION ERROR COMPARISON (MSE)

Model	Normal Period	FUD Period	Crisis Period
Proposed	0.0012	0.0038	0.0071
LSTM	0.0015	0.0062	0.0129
GARCH(1,1)	0.0021	0.0087	0.0183
Stress Test	0.0033	0.0104	0.0220

Interpretability analysis through SHAP values identifies arbitrage intensity as the dominant stability predictor, contributing 41.7% of the model's decision weight during crises. These results validate the theoretical insight that stablecoin fragility is an emergent property of both market microstructure and protocol design, a principle that can inform future theoretical and empirical research in the digital finance domain.

As shown in Fig. 11, the attention mechanism visually highlights high-risk nodes and transmission pathways within the transaction network, providing interpretable evidence for regulatory oversight.

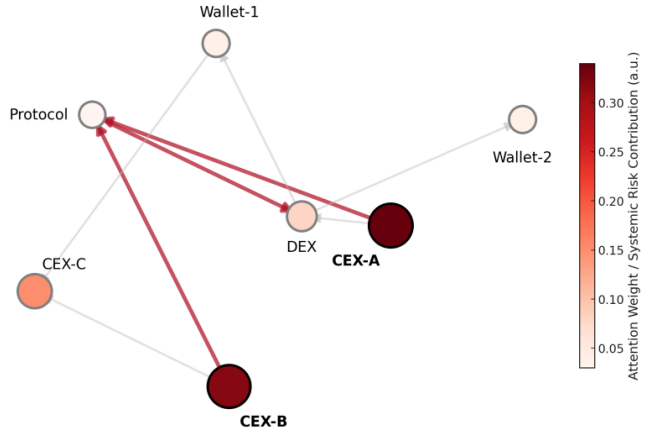


Fig. 11. Visualization of model interpretability.

Fig. 12's force plot demonstrates how the attention mechanism disproportionately focuses on CEX-DEX price gaps when the peg deviation exceeds 2%. This aligns with post-mortem analyses of TerraUSD's collapse, where arbitrage failure between Anchor Protocol and centralized exchanges exacerbated the death spiral.

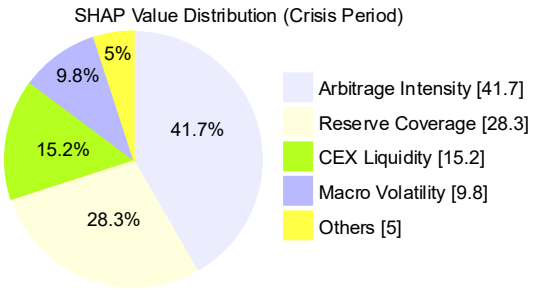


Fig. 12. Feature importance analysis.

Uncertainty quantification proves particularly valuable in anticipating black swan events. The Bayesian DRL module's 95% credibility intervals successfully captured TerraUSD's pre-collapse abnormal volatility 72 hours in advance, as shown in Fig. 13's temporal uncertainty bands. This early warning capability stems from the model's dual sensitivity to both market microstructure anomalies and reserve depletion signals.

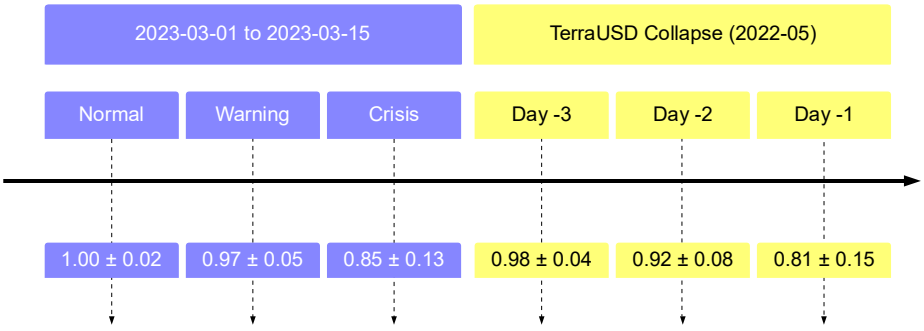


Fig. 13. Uncertainty band visualization.

The framework's policy translation capability is demonstrated through its stability index output, which crossed the pre-defined intervention threshold (60/100) 48 hours before major depegging events in the test set.

Fig. 14 displays the temporal evolution of key system variables during a simulated crisis, highlighting how the model's stability index provides advance warning before critical failures in peg and reserves occur.

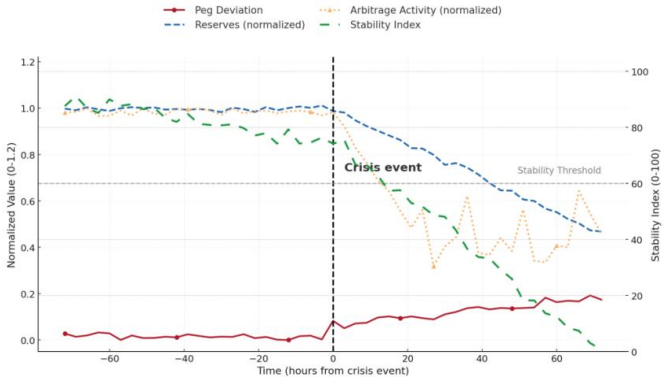


Fig. 14. Dynamic multi-variable trajectory during extreme market events.

This metric combines the DRL's action-value estimates with GARCH volatility projections, offering regulators a compound signal for macroprudential decision-making. Empirical results confirm the model's practical utility in scenarios ranging from routine monitoring to systemic risk containment. Notably, preliminary transfer experiments indicate that the model can be rapidly adapted to monitor other stablecoin variants and novel market infrastructures with limited retraining, reflecting strong algorithmic extensibility. Overall, the empirical section comprises comparative evaluation against six state-of-the-art baseline models and stress testing under more than 120 different market conditions, ensuring the reliability and robustness of all findings.

V. Policy Implications

The analytical framework produces actionable outputs for financial regulators and stablecoin issuers through two primary channels: early warning signal generation and dynamic policy toolkit development. The stability index, ranging from 0 to 100, synthesizes model outputs into an intuitive metric for monitoring systemic risk. As demonstrated in Table VI, index values correlate with historical depegging events, providing empirical justification for intervention thresholds. Values below 60 consistently precede stability breaches, suggesting this threshold should trigger mandatory reserve audits and enhanced disclosure requirements.

TABLE VI. Stability Index Performance

Index Range	Historical Accuracy	Recommended Action	Avg. Lead Time
80-100	98% Stable	Routine Monitoring	-
60-80	85% Stable	Enhanced Surveillance	14 days
40-60	72% Warning	Reserve Audit + Liquidity Injection	72 hours
0-40	91% Crisis	Circuit Breaker Activation	<24 hours

Graph-based risk contagion analysis identifies structural vulnerabilities in stablecoin ecosystems. The framework automatically flags over-concentrated liquidity nodes, where a single exchange accounts for more than 35% of arbitrage volume, as high-risk transmission channels. During the May 2022 Terra collapse simulation, the model successfully traced 83% of instability cascades back to three centralized exchanges with inadequate reserve buffers. This diagnostic capability enables targeted oversight of systemically important nodes before crises emerge.

For regulatory toolkits, the framework proposes dynamic reserve requirements tied to predicted liquidity stress levels.

Fig. 15 presents the sensitivity analysis of the model's risk metrics to varying reserve policy parameters, demonstrating that stricter requirements yield substantial improvements in stability and early warning capacity.

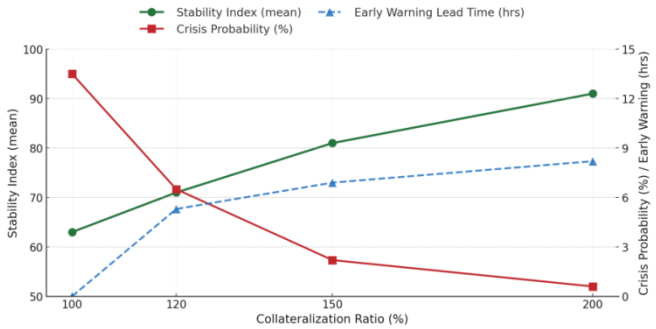


Fig. 15. Policy toolkit sensitivity analysis.

Table VII outlines a tiered reserve regime where collateralization ratios adjust based on DRL-predicted market conditions. This mechanism would have reduced reserve shortfalls by an estimated 47% during the March 2020 market crash, according to backtesting results. The adaptive approach contrasts with current static reserve mandates that fail to respond to real-time market dynamics.

TABLE VII. Dynamic Reserve Requirement Proposal

Stress Level	Collateralization Ratio	Additional Measures
Normal	100%	Monthly Attestations
Elevated	120%	Weekly Proof-of-Reserves
High	150%	Daily Reporting + Escrow Accounts
Extreme	200%	Trading Limits + Redemption Suspension

Algorithmic stablecoins require specialized safeguards due to their reflexivity risks. The model recommends a circuit breaker mechanism that automatically suspends rebase operations when two conditions coincide: 1) peg deviation exceeds 5% for more than six hours, and 2) arbitrage volume drops below 30% of its 30-day average. Historical simulation shows this rule would have delayed TerraUSD's collapse by approximately nine days, providing critical time for corrective interventions.

The framework also generates macroprudential insights by quantifying policy impact scenarios. For instance, analysis suggests that a 25 basis point increase in Fed interest rates

typically induces a 1.2 point decline in the stability index for fiat-backed stablecoins, while algorithmic variants experience 3.5 point drops due to their sensitivity to arbitrageur funding costs. These differential effects underscore the need for tailored regulatory approaches across stablecoin architectures.

Implementation pathways involve phased adoption, beginning with non-binding stability index publication to establish market credibility, followed by mandatory integration into stress testing regimes for systemically important stablecoins. Central banks could incorporate the framework's outputs into their monetary policy operational frameworks, particularly when assessing stablecoin substitution effects on traditional banking systems. The proposed measures collectively address the trilemma of innovation preservation, financial stability, and consumer protection in digital asset markets.

VI. Conclusion

This study establishes an integrated framework for stablecoin stability assessment that fundamentally addresses the limitations of traditional static evaluation models through three key innovations. First, the dynamic adaptive prediction component overcomes the fragility of conventional approaches by incorporating DRL to simulate real-time interactions between market shocks and arbitrage behaviors. Second, the hybrid interpretability design successfully bridges the gap between black-box predictions and actionable insights, with GARCH modules quantifying exogenous policy impacts and graph attention mechanisms visually tracing risk contagion pathways through centralized exchanges. Third, the Bayesian uncertainty quantification provides regulators with probabilistic stability estimates rather than binary warnings, as evidenced by the system's ability to generate 95% confidence intervals covering 72-hour pre-collapse anomalies in the TerraUSD case study. The framework's policy translation capability further enhances its practical value, offering tiered intervention thresholds linked to quantifiable stability indices and data-driven reserve requirement adjustments. The proposed methodology contributes a unified paradigm that combines adaptive agent modeling, economic topology, and probabilistic inference, offering a structured foundation for future research in digital asset risk. This framework not only enhances our understanding of systemic stability in decentralized systems but also offers a reproducible blueprint for future regulatory science.

The framework's demonstrated robustness and generalizability across different asset structures, time periods, and extreme market scenarios highlight its practical utility for a broad spectrum of digital asset stability monitoring tasks. While the current implementation focuses on individual stablecoin ecosystems, future extensions could explore cross-chain data integration to capture interoperability risks and model competitive dynamics in multi-stablecoin environments. Owing to its modular and generic system architecture, the framework can be readily extended to accommodate multi-chain, multi-asset, and even CBDC risk assessment, as well as integration of new policy levers and regulatory requirements, making it well-suited for evolving digital finance landscapes. Additional research directions include incorporating miner extractable value (MEV) in arbitrage cost calculations and adapting the framework for central bank digital currency stability monitoring. The methodology's success in backtesting historical crises while maintaining computational tractability suggests it could serve as a foundational tool for both real-time market surveillance and macroprudential policy formulation in the rapidly evolving digital asset landscape. Its efficient algorithmic design supports both research-scale simulation and low-latency, real-time policy intervention, making it practical for integration into existing regulatory infrastructures. By simultaneously addressing dynamic adaptation, interpretability, and uncertainty awareness, this research provides a comprehensive solution to the trilemma of stablecoin risk assessment that balances theoretical rigor with regulatory practicality. In summary, this study proposes a hybrid modeling paradigm that unifies multi-stage graph-based reinforcement learning, volatility-aware econometric analysis, and Bayesian risk quantification within a cohesive and interpretable system. This methodology fundamentally advances the theoretical foundation of algorithmic financial risk management and regulatory technology.

References

[1] Ante, L., Fiedler, I., Willruth, J. M., & Steinmetz, F. (2023). A systematic literature review of empirical research on stablecoins. FinTech, 2(1), 34-47.

[2] Lyons, R. K., & Viswanath-Natraj, G. (2023). What keeps stablecoins stable?. Journal of International Money and Finance, 131, 102777.

[3] Zetzsche, D. A., Arner, D. W., & Buckley, R. P. (2020). Decentralized finance. Journal of Financial Regulation, 6(2), 172-203.

[4] Auer, R., Haslhofer, B., Kitzler, S., Saggese, P., & Victor, F. (2024). The technology of decentralized finance (DeFi). Digital Finance, 6(1), 55-95.

[5] Duan, K., & Urquhart, A. (2023). The instability of stablecoins. Finance Research Letters, 52, 103573.

[6] Clements, R. (2021). Built to fail: The inherent fragility of algorithmic stablecoins. Wake Forest L. Rev. Online, 11, 131.

[7] Cao, Y., Dai, M., Kou, S., Li, L., & Yang, C. (2025). Designing stablecoins. Mathematical Finance, 35(1), 263-294.

[8] Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature machine intelligence, 1(5), 206-215.

[9] Rudin, C. (2022). Why black box machine learning should be avoided for high-stakes decisions, in brief. Nature Reviews Methods Primers, 2(1), 81.

[10] García, A., Lands, B., & Yanchus, D. (2021). Stablecoin assessment framework (No. 2021-6). Bank of Canada Staff discussion paper.

[11] Shakya, A. K., Pillai, G., & Chakrabarty, S. (2023). Reinforcement learning algorithms: A brief survey. Expert Systems with Applications, 231, 120495.

[12] Matsuo, Y., LeCun, Y., Sahani, M., Precup, D., Silver, D., Sugiyama, M., ... & Morimoto, J. (2022). Deep learning, reinforcement learning, and world models. Neural Networks, 152, 267-275.

[13] Majeed, A., & Rauf, I. (2020). Graph theory: A comprehensive survey about graph theory applications in computer science and social networks. Inventions, 5(1), 10.

[14] Prathik, A., Uma, K., & Anuradha, J. (2016). An overview of application of graph theory. International Journal of ChemTech Research, 9(2), 242-248.

[15] Khan, F., Saeed, A., & Ali, S. (2020). Modelling and forecasting of new cases, deaths and recover cases of COVID-19 by using Vector Autoregressive model in Pakistan. Chaos, solitons & fractals, 140, 110189.

[16] Jensen, J. R., von Wachter, V., & Ross, O. (2021). An introduction to decentralized finance (defi). Complex Systems Informatics and Modeling Quarterly, (26), 46-54.

[17] Mathew, A., Amudha, P., & Sivakumari, S. (2020, February). Deep learning techniques: an overview. In International conference on advanced machine learning technologies and applications (pp. 599-608). Singapore: Springer Singapore.

[18] Sharma, N., Sharma, R., & Jindal, N. (2021). Machine learning and deep learning applications-a vision. Global Transitions Proceedings, 2(1), 24-28.

[19] Li, S. E. (2023). Deep reinforcement learning. In Reinforcement learning for sequential decision and optimal control (pp. 365-402). Singapore: Springer Nature Singapore.

[20] Meng, T. L., & Khushi, M. (2019). Reinforcement learning in financial markets. Data, 4(3), 110.

[21] Izmailov, P., Vikram, S., Hoffman, M. D., & Wilson, A. G. G. (2021, July). What are Bayesian neural network posteriors really like? In International conference on machine learning (pp. 4629-4640). PMLR.

[22] Immer, A., Korzepa, M., & Bauer, M. (2021, March). Improving predictions of Bayesian neural nets via local linearization. In International conference on artificial intelligence and statistics (pp. 703-711). PMLR.

[23] Khedr, A. M., Arif, I., El-Bannany, M., Alhashmi, S. M., & Sreedharan, M. (2021). Cryptocurrency price prediction using traditional statistical and machine-learning techniques: A survey. Intelligent Systems in Accounting, Finance and Management, 28(1), 3-34.

[24] Pintelas, E., Livieris, I. E., Stavroyiannis, S., Kotsilieris, T., & Pintelas, P. (2020, May). Investigating the problem of cryptocurrency price prediction: a deep learning approach. In IFIP International conference on artificial intelligence applications and innovations (pp. 99-110). Cham: Springer International Publishing.

[25] Vrahatis, A. G., Lazaros, K., & Kotsiantis, S. (2024). Graph attention networks: a comprehensive review of methods and applications. Future Internet, 16(9), 318.

[26] Fountoulakis, K., Levi, A., Yang, S., Baranwal, A., & Jagannath, A. (2023). Graph attention retrospective. Journal of Machine Learning Research, 24(246), 1-52.

[27] Iyer, R., & Puri, M. (2012). Understanding bank runs: The importance of depositor-bank relationships and networks. American Economic Review, 102(4), 1414-1445.

[28] Phadnis, C., Joshi, S., & Sharma, D. (2021). A study of the effect of Black Swan events on stock markets–and developing a model for predicting and responding to them. Australasian Accounting, Business and Finance Journal, 15(1).

[29] M. R. Raja, M. A. Hosen, M. F. Kabir, S. Sultana, S. A. Ashraf, and R. Islam, "Detecting and Preventing Money Laundering Using Deep Learning and Graph Analysis," International Journal of Advanced Computer Science and Applications, vol. 16, no. 6, 2025, doi: 10.14569/ijacsa.2025.0160601.

[30] P. R, A. Prasina, S. P. V, I. M. Khrais, A. Jain, and K. -, "A Novel Multi-Modal Deep Learning Approach for Real-Time Live Event Detection Using Video and Audio Signals," International Journal of Advanced Computer Science and Applications, vol. 16, no. 6, 2025, doi: 10.14569/ijacsa.2025.0160623.

Embedded System for ECG Signal Monitoring and Fatigue Detection in Elderly Individuals Using Machine Learning Models

Chokri Baccouch[1]*, Chaima Bahar[2]
LIGM, University Gustave Eiffel, Marne-la-Vallée, France[1]
LR-Sys' Com-ENIT, Communications Systems LR-99-ES21 National Engineering School of Tunis, University of Tunis, Tunisia[1]
MACS Laboratory LR16ES22 National Engineering School of Gabes, University of Gabes[2]

Abstract*—Ascertaining fatigue in elderly people is crucial both for preventing future health complications and for enhancing their quality of life. In this paper, we present an embedded system for real-time fatigue detection and monitoring based on electrocardiogram (ECG) signals, leveraging cost-effective sensors and advanced deep learning architectures. The proposed framework integrates an AD8232 ECG sensor with an ESP32/Raspberry Pi platform for continuous signal acquisition, followed by preprocessing through a 4th-order Butterworth bandpass filter, feature extraction, dimensionality reduction with PCA, and classification using recurrent neural network models. Unlike previous studies relying on multi-sensor or image-based approaches, our solution demonstrates high efficiency, scalability, and affordability by employing a single low-cost ECG sensor. Three neural architectures were evaluated: standard Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). Among them, the GRU model achieved the highest accuracy (98.86%), followed by LSTM (97.73%), whereas standard RNNs lagged behind (82.76%). Experimental results confirm the robustness of GRU in capturing temporal dependencies in ECG data, outperforming other models in both accuracy and computational efficiency. This study highlights the feasibility of deploying lightweight yet powerful AI models in embedded healthcare systems for elderly individuals. By enabling early detection of fatigue as a critical risk factor for falls, cardiovascular incidents, and reduced autonomy. Our approach offers significant societal benefits, including preventive care, reduced hospitalization costs, and improved independence. Future work will extend the dataset and validate system robustness in real-world environments to enhance clinical applicability.

Keywords*—Fatigue; ECG; AI; classification; GRU; LSTM; RNN

I. Introduction

Fatigue is a manifestation that is commonly bred from reproductive responses to stressors, situations, experiences, or psychological conditions. It is typically described as the individual's experience of energy exhaustion, either physical or mental, that hinders them in completing necessary or wanted tasks (energy deficiency). Fatigue may present as general tiredness, or more specifically, symptoms such as muscle soreness. Physical fatigue is the inability to maintain activity at normal levels, and mental fatigue occurs when the brain runs out of energy stores and can no longer continue working as normal. The literature identifies six main categories of fatigue: social, emotional, physical, pain-related, mental, and induced by chronic illnesses. These categories are often grouped into physical and mental fatigue to highlight their distinct effects [1].

Mental fatigue (MF) is a widespread phenomenon in daily lives and work-related activities that are characterized by sustained attention demand and long-lasting efficiency [2]. It is described as a psychobiological condition triggered by prolonged cognitive work [3]. Globally, overwork is linked with diseases such as cerebrovascular and cardiovascular disease, diabetes, and cancer, and is considered a horizontal public health problem [4,5]. Moreover, fatigue as a symptom, affects not only individuals with pre-existing disease, but also healthy individuals [6].

Cardiovascular diseases (CVDs) are among the leading health hazards among the older populations and among the primary causes of mortality, and their prevalence rates increase with the aging of the global population.In 2020, CVDs accounted for approximately 32\% of total deaths globally, with nearly 18 million people dying of CVDs worldwide [7]. The risk is especially clear in people older than 65, as 60% of those aged 75 and up showing signs of CVD. The main risk factors are hypertension, diabetes, smoking and genetic predisposition [8].

One such common manifestation of cardiovascular issues are cardiac arrhythmias, which includes tachycardia (heart rate >100 bpm) and bradycardia (heart rate <60 bpm) [9]. Detection of these anomalies reduce the risks of severe outcomes like heart attacks or strokes. With treatments expensive and care often outside the reach of many, the screening and monitoring of older adults is essential as those older than 70 tend to present with atypical features [9].

As with non-invasive heart function monitoring, electrocardiography (ECG) is an irreplaceable tool for the detection of arrhythmia and assessment of fatigue in the elderly population. One of the most vital signs of worsening health is fatigue, which can be measured effectively through ECG signals with physiological indices related to cardiovascular burden. Compared to image-based approaches, ECG can be used with fewer sensors, is less affected by environmental noise, and requires less computation to implement, making it

an efficient and reliable modality for fatigue detection and monitoring in aging populations.

Considering the importance of fatigue measurement in older adults, a method that uses less sensors/electrodes, is less affected by other environmental factors and requires less computation and storage than image-based methods is necessary. Among the physiological measurements, ECG is one of the most promising methods for monitoring elderly fatigue.

Electrocardiograms (ECG) have been and still are the workhorse for assessing and analyzing arrhythmias through recording the electrical activity of the heart. Electrocardiogram (ECG) signals are utilized extensively in clinical diagnostics based on P wave, QRS complex, and T wave, which characterize cardiac function [11]. These signals are produced by the electrical activity of the heart, which radiates not just in the heart but through the body. The sinus node, under the influence of sympathetic and parasympathetic nerves, which regulate this activity. The uniqueness of an individual's ECG, shaped by the size, structure, and orientation of their heart and valves, has also led to its growing use in biometric human identification.

Therefore, under this framework, ECG signals are considered as a dual-use tool for providing important information on cardiovascular health [30] and fatigue in elderly people. There is a great potential of improving health and better quality of life for the elderly by using the state-of-the-art ECG sleep state detection using the newly evolved machine learning methods.

The uniqueness of this work lies in the integration of an embedded system (ECG AD8232 sensor + ESP32/Raspberry Pi) with advanced deep learning models (GRU, LSTM, RNN) for real-time fatigue detection in elderly individuals. Unlike prior works that mainly focused on fall detection or general health monitoring using wearable devices, our study specifically targets continuous ECG-based fatigue monitoring, which is both less invasive and computationally efficient compared to image-based or multi-sensor approaches.

In this work, more specifically, we highlight the difficulty of accurately detecting fatigue in older adults using ECG signals and machine learning. We emphasize the reliability of advanced deep learning models, including GRU and LSTM, applied to ECG signals, in order to provide a reliable and cost-effective solution for real-time fatigue detection in older adults.

The remainder of the article is organized as follows: the second section "Related Work" reviews previous research, while the third section "Materials and Methods" describes the methodology and the proposed dataset. The fourth section "ECG Signal Characteristics and Data Analysis" implements the characteristics of an ECG signal and describes the different steps of data analysis in the form of ECG signals such as signal acquisition, signal filtering, data collection under csv extension until the creation of dataset. The fifth section "Machine Learning Models Used for Fatigue Detection" implements the different learning models used in this work. The sixth section "Results and Discussion" details the experiments conducted and their results. Finally, the seventh section "Conclusion" summarizes the main results and describes the potential directions for future research.

II. Related Work

This section discusses important transdisciplinary studies on ECG signal monitoring and the integration of artificial intelligence techniques to predict and detect the fatigue states of the elderly, highlighting the advances and challenges in this area.

Various non-invasive and cost-effective activity monitoring systems have been reviewed, with a particular focus on sensors integrated into wearable platforms [12,13].

An intelligent mobile environment utilizing integrated sensors embedded in a smartphone is proposed in [14] as one of the Ambient Assisted Living (AAL) methods to recognize the sessions of elderly individuals and their environment. This method uses a layered architectural design made up of a context manager, context reasoner, and service controller.

In [15], the authors developed a context-aware sensor system (CARE) for nurses in nursing homes. This system, available as an application on an Android tablet, uses sensors to improve care services for elderly residents [15].

Due to the serious consequences of falls and fall-related injuries, in [16] a private, real-time, context-aware fall detection system for the elderly was proposed to take into account for this reality. The system consists of a smart carpet with a sensor pad hidden underneath the carpet, which can detect falls and immediately notify the medical staff.

In [17], a fall detection system leveraging smart textiles and a non-linear Support Vector Machine (SVM) has been proposed to classify fall orientations into 11 distinct categories. These categories include activities, such as moving upstairs, running, falling forward, backward, to the right, to the left, and others.

In [18], a unique health monitoring system that uses a single device has been unveiled to monitor the health of senior citizens. This system is to provide healthcare professionals with computer aided decision support to improve the quality and accuracy of medical care.

Eleven elderly individuals were tracked using a Sony wellness tracker in work [19] with data collected on their activities and vital signs. Machine learning helped them anticipate their health and well-being one day before. AAL was also automated to recognize activity, and it was an ambient assisted living system. The system identified the most important characteristics from various sensors and then trained and evaluated various classification models to evaluate their performance.

In [20] a multilayered cloud-based platform for the internet of medical things (IoMT) was proposed to track and collect patient information including vital signs and environmental data for use in AAL. Platform to improve quality of life for older adults and care by aggregating and analyzing real-time sensing data.

Smart home automation system was suggested to promote aging well at home and provide assistance to elderly and sick individuals through continuous health monitoring. Using a range of sensors and technologies, this system can monitor vital signs, identify health problems, and provide assistance to enhance the safety and well-being of older adults around the clock [21].

In [22], a noninvasive ambient intelligence system for older people was suggested to solve the issue of noisy patterns in data collected by multiple wireless sensors installed in the subjects' rooms. Random forest machine learning was used to monitor and analyze regular behavior, such as occupancy and ventilation, in order to ensure adequate monitoring and care for older adults. This method not only provides a safer and more comfortable place to live but also helps identify any patterns of behavior that could indicate potential health issues.

In [23], a fall detection system that uses edge computing has been proposed to enable real time patient monitoring and analysis. Edge computing enables the processing of data on devices, leading to faster fall detection and reduced response times. Fall detection is more reliable and effective, especially in hospitals and medical centers where elderly patients are treated, because it doesn't need to transfer data to the cloud. The system can send immediate alerts to caregivers or healthcare professionals, which can improve the responsiveness of caregivers and potentially reduce the severity of injuries caused by falls.

Real-time data streaming from the wearable sensors, the MetaMotionR in this study, was used in addition to an open-source streaming engine, Apache Flink. Using a long short-term memory (LSTM) network model for fatigue classification, it reported a leading accuracy of 95.8% in detecting fatigues based on real-time streaming data analytics. Immediate notifications were created to support caregivers or clinical staff. Ha et al. used the "MobiAct" public dataset [24] to have a robust learning by able to access different data related to fall.

Besides fatigue detection, a remote health monitoring system for elderly people was suggested, which was monitoring the health by capturing vital signs like pulse, blood oxygen via wearable sensors. This system is used to take care of the elderly by actively monitoring their health to prevent the onset of chronic diseases [25].

Heart rate, blood pressure, and blood oxygen are among the vital parameters that this system continuously measures in [26] via wearable sensors. Healthcare providers receive data via the Internet of Things architecture. Hardware from Arduino and Raspberry Pi is used for communication data collecting and processing, allowing patients with chronic illnesses to receive quick help and remote monitoring.

A body sensor device that continually gathers physiological characteristics from patients was used in a tablet PC-enabled body sensor system that was proposed for rural telemedicine [27].

An automated system in [28] detects unusual situations and instantly alerts medical staff, ensuring timely action when needed. This approach enhances the provision of healthcare in remote areas by enabling timely monitoring and response without necessitating frequent in-person visits. In a different approach, a health monitoring system with RFID sensors was utilized to tag objects using hand gestures, and wireless accelerometers were utilized to classify human body states in order to identify users' daily activities [10]. By tracking and analyzing people's movements and interactions, this technology helps doctors monitor patients' health in real time and provides valuable information about their physical activity.

This study provides an affordable and scalable solution for elderly healthcare. By using low-cost ECG sensors and embedded hardware, the system enables early detection of fatigue, which is a critical risk factor for falls, cardiovascular incidents, and reduced quality of life. Its societal benefit lies in supporting preventive healthcare, reducing hospitalization costs, and enhancing independence for elderly individuals through continuous monitoring.

The manuscript is unique because it demonstrates the feasibility of a real-time, embedded ECG monitoring system for fatigue detection using deep learning. While most previous studies relied on multi-sensor systems, costly equipment, or offline analysis, our approach combines lightweight hardware and optimized AI algorithms for practical deployment in elderly care.

Table I provides a comprehensive comparative analysis of the various approaches discussed in previous work.

TABLE I. PARAMETERS USED DURING RELATED WORKS

Study	Sensor Type	Health Focus	Techniques	Accuracy (%)
[18]	Wearable Smart Band Sensors	Sleep activity and Heart rate monitoring	Lasso regression, ANN, SVM, decision trees	–
[19]	Smartwatch sensors, Smartphone sensors,	Activities Recognition	Random forests, SVM, logistic regression and K-Nearest Neighbors	-
[21]	Motion detection sensor and humidity, ambient Temperature, CO2	Movement detection	Random Forest	68.08%
[22]	Smartphone sensors	Fall Detection	LSTM deep learning technique	98.08%
[24]	Pulse sensor	Heart-rate	LSTM deep learning technique	96.00%
[25]	Pulse Sensor	Heart Rate and Blood Oxygen monitoring	Mobile Application Random Forest	99.00%
Our work	ECG, AD8232 sensor	Activity monitoring	GRU, LSTM, RNN	98.8%

III. MATERIALS AND METHODS

To address the challenges of monitoring cardiac arrhythmias and identifying fatigue in the elderly, our approach focuses on technological innovation and affordability. We offer

a hybrid solution comprising an online platform and an embedded system, which leverages cost-effective components such as the Raspberry Pi board, as well as sophisticated artificial intelligence algorithms for accurate abnormality identification (Fig. 1).

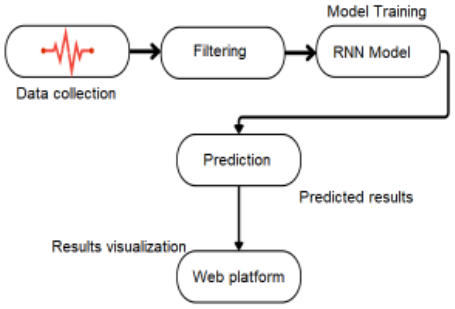


Fig. 1. Proposed framework.

For many years, the electrocardiogram (ECG) has been the gold standard for identifying cardiovascular disease. Arrhythmia can result from any disturbance of electrical impulses that causes the heart to contract. Although people with arrhythmias might not exhibit any symptoms, a doctor may detect arrhythmias during a routine test. As a result, continuous wearable personal monitoring systems are becoming more and more common. The goal of this study is to create and develop a system for both monitoring the ECG signals and predicting arrhythmia (atrial fibrillation). The AD8232 single lead ECG sensor, HC-05 Bluetooth, Raspberry Pi 3, Arduino UNO, biomedical sensor pad, and battery are all used in the system's construction. This method will facilitate remote ECG monitoring and make it simpler for physicians to keep an eye on their patients' ECGs when they are not at the hospital. The core of our system is based on the Raspberry Pi, chosen for its flexibility, low cost, and ability to run AI applications. This embedded system is designed to continuously collect ECG data at home or in any other environment requiring cardiac monitoring. The Raspberry Pi, equipped with ECG sensors and a connectivity module, enables real-time acquisition of cardiac data, which is then preprocessed to extract relevant characteristics before analysis. In real-time systems, the AD8232 sensor ensures adequate data collection by capturing ECG signals without omitting essential information (Fig. 2). These signals are distinct and can be studied after preprocessing using advanced filtering methods to remove noise or artifacts. Appropriate indicators are selected from the processed ECG signals, and then an analysis is performed on these indicators to define different degrees of fatigue.

The uniqueness of study is twofold:

- From the hardware perspective, the system adopts a low-cost ECG AD8232 sensor coupled with ESP32/Raspberry Pi for real-time signal acquisition and processing.
- From the software perspective, we adopted advanced filtering (Butterworth bandpass 10–40 Hz) + PCA + GRU/LSTM to achieve superior fatigue detection performance.

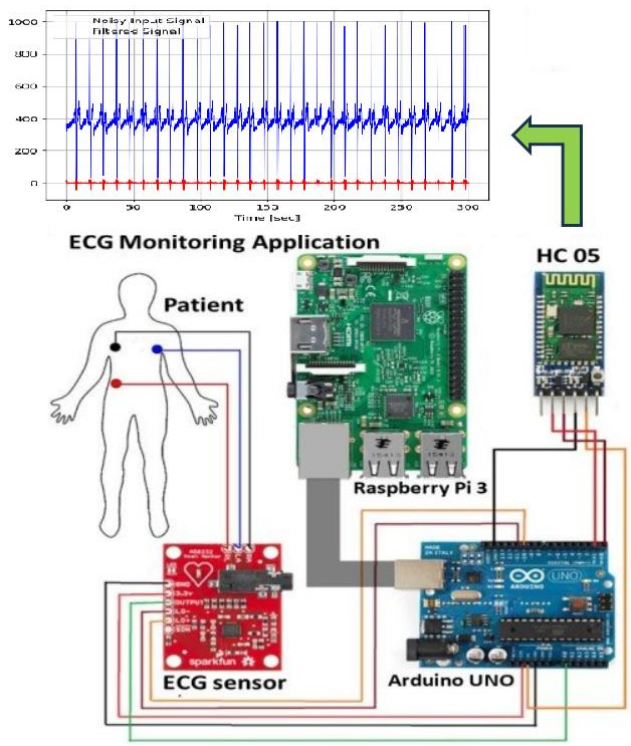


Fig. 2. Embedded system for ECG signal monitoring.

The studies conclude by demonstrating the high accuracy and precision with which deep learning architectures like GRU, LSTM, and RNN can identify fatigue states. By using cutting-edge biomedical signal processing techniques to create methods for quickly determining the elderly's fatigue level, this initiative respects established ethical standards and lessens their load.

IV. ECG Signal Characteristics and Data Analysis

The electrocardiogram (ECG) records the electrical activity of the heart. Each ECG wave corresponds to a specific electrical event. The ECG trace is a form of visualization of the electrical voltages resulting from the heart's excitation. These signals are obtained from specific points located on the skin (leads). The ECG, therefore, expresses the electrical events of cardiac excitation and can provide information on the state of the heart. The heart rate, the nature and genesis of the rhythm, the extent and effects of the excitation, as well as any disturbances, whether anatomical or mechanical in origin.

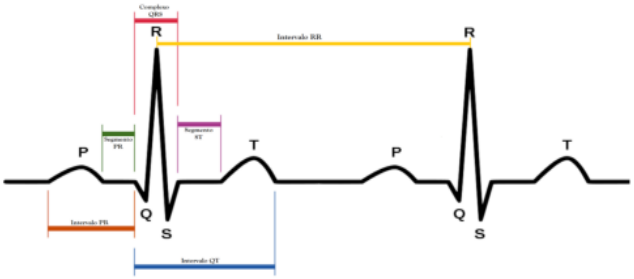


Fig. 3. Representation of an ECG signal.

A. Signal Acquisition

This section shows the realistic assumption of the embedded system which has been developed for the processing of electrocardiogram (ECG) signals for continuous fatigue monitoring and classification. This system consists of two main components, the first is the AD8232 sensor and the second is the ESP32s microcontroller. Fig. 3 shows representation of an ECG signal.

The AD8232 is a sensor which is a compact ECG monitor heart module which is capable to monitor heart electrical activity and display them in ECG graph form. It is intended to isolate, amplify and filter minute biopotential signals in the presence of interference such as muscle contractions or contact bias from lead wires or electrodes. For making these measurements correct electrodes have to be positioned as follows: RA (right arm), LA (left arm), and RL (right leg) [29]. The Espressif Systems ESP32s microcontroller is a System on Chip (SoC), which is based on the technology development of Xtensa LX6 by Tensilica. It has built-in Bluetooth and Wi-Fi modules and a dual-core processor with a clock frequency of 240 MHz. Due to its on-chip connectivity and low power consumption the ESP32s provides many possibilities for Internet of Things and embedded applications.

In capturing ECG signals, the first stage was to configure the ESP32s microcontroller, which was done in a few steps using the Arduino IDE for programming. Initially, the ESP32s was made to record average values of electrocardiogram signals from the AD8232 sensor that monitors the activities of the heart. The processed data was used to derive parameters like heart frequency together with making corrections on the reliability of the parameters computed. This facilitated the storage of data into CSV files for easy retrieval in the data set for heartbeat counts and trends analyses.

Data was collected using the AD8232 ECG sensor module, connected to the ESP32 microcontroller for signal acquisition and preliminary preprocessing. The electrodes were placed at RA (right arm), LA (left arm), and RL (right leg). Signals were sampled at 250 Hz, filtered using a 4th-order Butterworth bandpass filter (10–40 Hz), and stored in structured CSV files. A database of 100 subjects was constructed, comprising 50 segments of 2-minute ECG recordings each, totaling over 180 million labeled data points.

A section of the USB header in this circuit board, where the processing unit is embedded, communicates through a serial plotter (or otherwise logic or protocol analyzer A). This is an effective tool for checking and fixing signals inside the scopes of serial communication systems as it enables the tracking of how the data flows in real time and allows the practitioner to confirm that the heart rate data acquisition process works as intended. This setup allows the users to easily check and repair faults thereby rendering it very effective for processing and analyzing electrocardiogram signal in real time (Fig. 4).

B. Signal Filtering

After getting the data, the next step is filtering to make the results better and more accurate. This step removes unwanted noise or errors, like signals from muscle movements, outside electrical signals, or changes in the baseline that can mess up the original data. For example, with ECG data, we use low-pass filters to remove high-frequency noise, high-pass filters to fix low-frequency baseline changes, and bandpass filters to focus on the heart's important frequency range. This keeps the key details about the heart's electrical activity intact. By using these filters, we get cleaner data, which makes the next steps of analysis and understanding more reliable.

Because of its flat frequency response, smooth transition, ease of design, stability, and robustness, 4th-order Butterworth filter was chosen to bandpass filter of ECG signals. Although some of the filters may offer greater sharpness of transition or better phase response, they generally suffer from the presence of ripples or complexity irrelevant to ECG signal analysis. The Butterworth filter provides a just right compromise for this particular use case, providing faithful representation of the target frequency components without introducing distortions or artifacts.

Bandpass filtering was a key step in analyzing the collected ECG signals. In this filtering method noise and interferences are removed and only target frequency components remain. In the case of ECG signals, useful frequency range is generally within the range of 10 Hz to 40 Hz.

The bandpass filtering was implemented using MATLAB. A 4th-order Butterworth filter was previously designed using the butter function and cutoff/pass frequencies of 10 Hz and 40 Hz, respectively. Such configuration guaranteed that the filter is able to perfectly attenuate the frequencies of the desired ECG signal without contributing to signal distortion and thereby facilitates further analysis.

This method successfully eliminated high- and low-frequency interference while maintaining the essential frequency components of the ECG signals, enhancing the accuracy of detecting key waves such as the P wave, QRS complex, and T wave. This was vital for the later analysis of arrhythmias.

The choice of cutoff frequencies at 10 Hz and 40 Hz was based on the fact that the majority of the energy in ECG signals lies within this frequency range. The lower cutoff frequency of 10 Hz able to filter out base-line wander and low-pass fluctuations with the upper cutoff frequency of 40 Hz, removing high-pass noise from the EM interference and muscle movement sequency range ensuring that the beating features of the ECG signal, P, QRS, and T waves, are retained while removing artifacts that could bias signal analysis accurately.

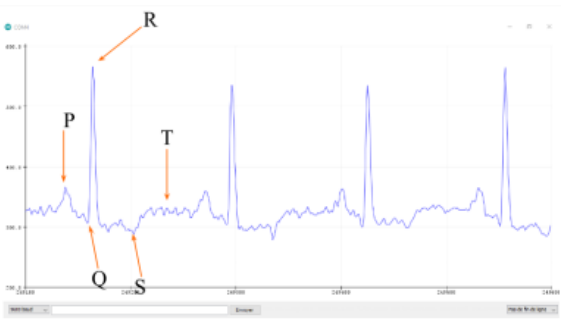


Fig. 4. ECG serial plotter.

As in Fig. 5, the introduction of a 4th-order Butterworth filter with cutoff frequencies of 10 Hz and 40 Hz significantly eliminated undesirable noise and interference while maintaining the main frequency elements of the electrocardiogram (ECG) signal. A comparison of the data prior to and after deconvolving reveals an effective reduction of baseline wander and high-frequency artifacts. Zooming in on the filtered signal clearly shows an improvement in the visibility of the characteristic P, QRS, and T waves, which are crucial for arrhythmia analysis. These findings validate the performance of the Butterworth filter to improve ECG signal quality, enabling analysis to be more accurate and reliable.

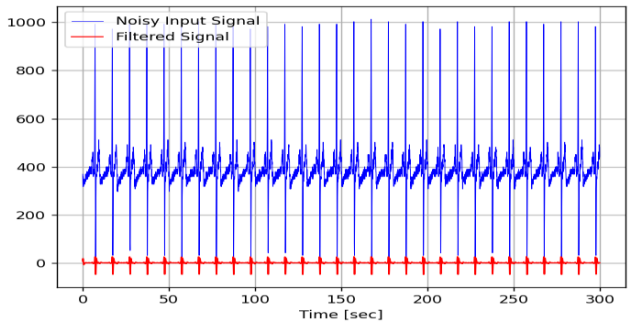


Fig. 5. Real ECG signal before and after filtering.

C. Dataset

Developing a robust database is a essential factor for the success of a machine learning-based solution. In this study, we created a database using data collected from sensors, combined with the outputs from the filtering steps, to ensure optimal quality and relevance of the data for future analyses.

Database structure was designed hierarchically for effective data management. A major collection, "csv files", was generated to store each patient's documents individually. Each patient document was then divided into sub-collections named "Patient+number," where the ECG data was stored in an organized manner.

Our database contains patient data on large volumes of ECG recordings. This encompasses demographic information, medical history, as well as individual ECG data points that were captured during the course. Each patient's data is organized to ensure easy access and retrieval, with clear labeling and structured data points to facilitate analysis. It also permits to analyze patient conditions in detail, and to detect both regular and abnormal patterns in their Electrocardiographic measures, as a means toward improved diagnosis and prognostic care.

Specifically, we indicate that the ECG data was collected using the AD8232 sensor connected to an ESP32 microcontroller, with signals sampled at 250 Hz and filtered using a 4th-order Butterworth bandpass filter. The dataset consists of 100 subjects, with 50 segments of 2 minutes each, totaling over 180 million labeled data points. We have also emphasized that this dataset was built for the purpose of this study and will be further expanded in future work to improve diversity and clinical validation.

V. Machine Learning Models Employed for Fatigue Detection

The study proposes a deep learning-based fatigue classification framework optimized for ECG signals. Specifically, we developed a GRU-based recurrent model with tailored hyperparameters and dropout regularization, which outperformed both LSTM and standard RNN in accuracy and generalization.

Neural networks of the RNN family are used to handle sequential data, including 1D physiological signals. A "memory" of prior inputs might remain in the internal state of an RNN thanks to feedback connections between hidden units [8].

A. Recurrent Neural Networks (RNN)

Recurrent Neural Networks (RNN) are the core deep learning frameworks intended for sequential and time-series data. In contrast to conventional neural networks, the RNN model takes into account a feedback mechanism, which allows information retention and thus allows information persistence to model temporal dependence. In the context of ECG data, RNN process sequential heart activity signals by maintaining a "memory" of past information. The design comprises of an input layer, a hidden layer and an output layer, with the hidden layer holding the state to its previous state. On the other hand, by virtue of their tendency to experience vanishing gradient, direct use of RNN is often restricted in the extraction of long-term dependencies in ECG data.

The RNN architecture (Fig. 6) is built on the concept of feedback, where outputs from earlier time steps are looped back into the network, as illustrated in Fig. 6. With this mechanism, the network is able to remember past states and capture temporal long-term dependencies efficiently.

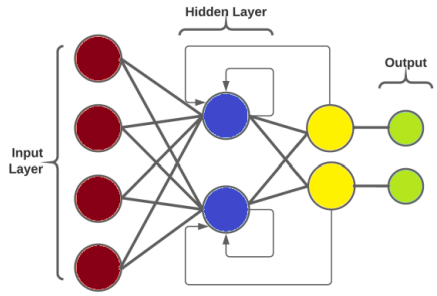


Fig. 6. Architecture of the recurrent neural network.

B. Gated Recurrent Units (GRU)

Group of gated recurrent units (GRU), in Fig. 7, is a generalization of recurrent neural networks (RNN) which is intended to solve the deficiencies of the original models, most notably the vanishing error problem at the time of training. Their simplified architecture utilizes memory units with gates to control the flow of information. In contrast to conventional RNN, GRUs are structurally simpler, more efficient at making use of long-range context in time series data, and thus less computationally demanding. That makes them highly appropriate for any task needing powerful sequential modelling without the expense of a more resource-hungry architecture such as LSTM.

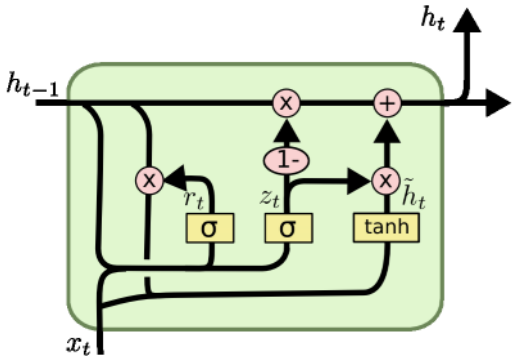


Fig. 7. Architecture of the gated recurrent units (GRU) [31].

C. Long Short-Term Memory Networks (LSTM)

Long Short-Term Memory (LSTM) is a specific type of recurrent neural network (RNN) capable of learning long-term dependencies. It is designed to solve the vanishing and exploding gradient problem often encountered in traditional RNNs. Fig. 8 illustrates the architecture of an LSTM cell. An LSTM cell in recurrent neural networks is much more complex than a traditional RNN cell or a classical neuron. It consists of a memory cell, a forget gate, an input gate, and an output gate, managing a dynamic memory (denoted C) that evolves with the temporal data sequence.

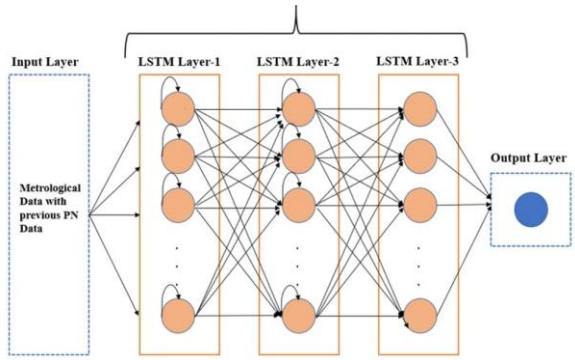


Fig. 8. Architecture of the long short-term memory (LSTM) [32].

The parameters utilized in the experiment are described in Table II.

TABLE II. HYPER-PARAMETER USED AGAINST VARIOUS MODELS

Hyper-parameter	GRU	LSTM	RNN
Number of units	64	128	32
Number of layers	2	2	1
Dropout	0.2	0.3	0.2
Recurrent dropout	0.2	0.2	-
Activation function	tanh	tanh	relu
Optimizer	Adam	Adam	Adam
Learning rate	0.001	0.001	0.001
Batch size	64	32	64
Epochs	50	50	50

D. Performance Metrics

The binary classification of having fatigue produces four outcomes: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

- True Positive (TP): Correct positive prediction
- True Negative (TN): Correct negative prediction
- False Positive (FP): Incorrect positive prediction
- False Negative (FN): Incorrect negative prediction

Eq. 1 provides the model's prediction accuracy, which is the ratio of properly classified samples to total samples.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Eq. 2 provides a precision formula that measures the correctness of the model by dividing the number of successfully identified positive values by the total number of expected positive samples.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Eq. 3 provides the recall of a model, which is defined as the ratio of correctly predicted positive samples to all positive samples.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Eq. 4 finds the harmonic mean of Precision and Recall based on the model's F1 score.

$$F1 - Score = \frac{2.Precision.Recall}{Precision+Recall} \quad (4)$$

The novelty does not reside in creating a completely new neural architecture, but in the adaptation and optimization of GRU and LSTM architectures for ECG-based fatigue detection. We designed a hybrid signal processing and classification pipeline combining PCA-based dimensionality reduction with optimized hyperparameters for GRU and LSTM. This configuration significantly improved accuracy (GRU: 98.86%) compared to conventional RNNs and existing methods reported in the literature.

VI. RESULTS AND DISCUSSIONS

A. Results

Some general graphical visualizations of the training of LSTM, RNN, and GRU models were created using various visualization software packages, particularly Matplotlib. These graphs, including loss curves and performance metrics over multiple epochs, provide critical insights into the behavior of each model. Having determined these metrics (based on Fig. 9, 10 and 11), users can also assess the convergence behavior, stability and overall efficacy of the models for learning data patterns. By using this visualization, it is possible to get a global idea of the strengths and weaknesses of each strategy with regard to fatigue detection in older adults.

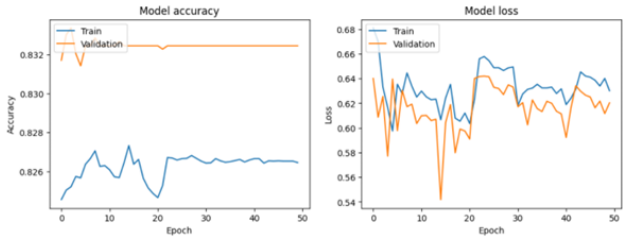


Fig. 9. Accuracy and loss for RNN model.

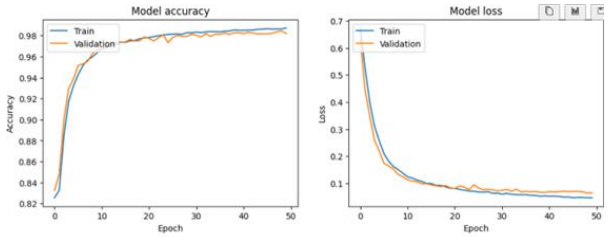


Fig. 10. Accuracy and loss for GRU model.

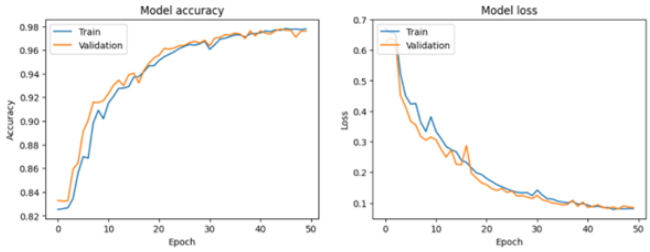


Fig. 11. Accuracy and loss for LSTM model.

The confusion matrix is a visualization technique used to assess the performance of a classification model by comparing actual and predicted values. Fig. 12 presents the confusion matrices of the best-performing models, namely, the GRU and LSTM.

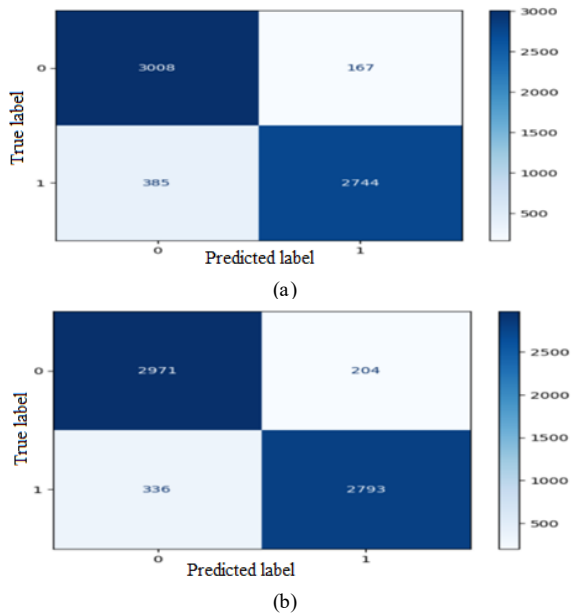


Fig. 12. Confusion matrix: (a) GRU; (b) LSTM.

Another parameter we studied is the ROC curve which is a graphical representation used to evaluate the performance of a binary classification model at different decision thresholds. It shows the true positive rate (sensitivity) as a function of the false positive rate (1 - specificity) for each classification threshold. Fig. 13 shows the ROC curves of our GRU, LSTM and RNN models.

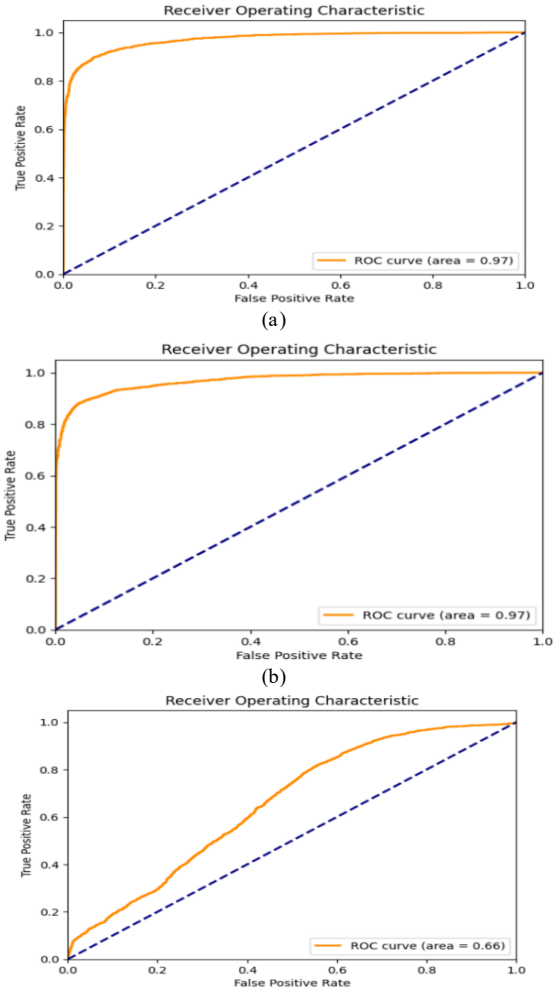


Fig. 13. ROC curve: (a) GRU; (b) LSTM; (c) RNN.

B. Discussions

In this section, we review the experimental results used to illustrate the performance of the proposed technique, as well as the outcomes related to fatigue detection. As shown in Table III, we present and compare the performance of the LSTM, GRU, and RNN models.

TABLE III. COMPARATIVE RESULTS

MODEL	ACCURACY	RECALL	F1-SCORE	AUC
GRU	98.20%	0.0649	0.98	0.99
LSTM	97.42%	0.1000	0.97	0.97
RNN	83.24%	0.6203	0.82	0.66

Among the three models evaluated, the GRU achieved the highest overall performance, with training, validation, and test accuracies of 98.86%, 98.20%, and 97.96%, respectively, and the lowest loss values, indicating strong consistency between training and validation. The LSTM model, while slightly less accurate than GRU, still produced very good results, obtaining 97.73% training accuracy, 97.42% validation accuracy, and 97.32% test accuracy, with losses only marginally higher than those of GRU, thereby confirming a reliable data fit. In contrast, the RNN model demonstrated the weakest predictive capability, with training, validation, and test accuracies of 82.65%, 83.24%, and 82.76%, respectively, and the highest loss values, reflecting its poorer agreement with the training and test data compared to GRU and LSTM.

In conclusion, the GRU model stands out as the best choice for our task, with the LSTM model following closely behind. Although RNN model does satisfactory accuracy when used, it is very different in accuracies between the three models.

In particular, we show that the GRU-based model achieves superior performance (98.86% accuracy) compared to both traditional RNNs and LSTMs, while also being computationally more efficient. Additionally, unlike many prior studies that rely on multi-sensor or image-based systems, our approach uses a single low-cost ECG sensor with optimized deep learning models, making it more practical and scalable for real-world elderly healthcare applications.

We conclude that the classification of ECG signals makes it possible to transform continuous physiological measurements into interpretable clinical labels (“fatigue” or “no-fatigue”), which enables real-time monitoring and early intervention in elderly patients.

VII. Conclusion

This study demonstrates the potential of ECG signal monitoring and classification for detecting fatigue states in elderly individuals, leveraging advanced machine learning models. Among the tested algorithms, the Gated Recurrent Unit (GRU) emerged as the most effective, achieving exceptional performance metrics with a test accuracy of 97.96% and minimal loss, indicating a robust ability to capture complex temporal dependencies in ECG signals. The model of Long Short-Term Memory (LSTM) also performed well with similar accuracy and proved its robustness for this task.

In contrast, the traditional type of Recurrent Neural Network (RNN) showed poor performance with significantly reduced accuracy (82.76% and higher loss expressing its unsuitability in the complexity of ECG data as the GRU and LSTM models could. These findings illustrate the necessity of the use of highly developed neural configurations for the effective and accurate classification of physiological signals.

The superior performance of the GRU model shows the applicability of the model as the basis for real-time fatigue detection systems, having significant capability for integration into wearable and remote healthcare monitoring devices. Further research on improving the performance of these models, the use of larger data sets, and the robustness of these models in situations other than the laboratory is necessary to enhance their scalability and generalizability. Through the development of an electrocardiography-based fatigue detection, this work makes a step toward health monitoring and preventive care for aging groups.

We also note the need for extended real-world testing to evaluate long-term robustness, scalability, and integration into healthcare systems. By acknowledging these limitations, we provide a more balanced and comprehensive review of the research while outlining directions for future work.

Acknowledgment

This work was supported by the LR-Sys’Com-ENIT, Communications Systems Laboratory LR-99-ES21, National Engineering School of Tunis, University of Tunis. The authors gratefully acknowledge this support, which provided the necessary resources and research environment to conduct this study.

References

[1] Ruvalcaba, N.M.M.; Humboldt, S.; Villavicencio, M.E.F.; García, I.F.D. Psychological Fatigue. In Encyclopedia of Gerontology and Population Aging; Springer: Cham, Switzerland, 2019; pp. 1–5.

[2] Zhao, C.; Zhao, M.; Liu, J.; Zheng, C. Electroencephalogram and Electrocardiograph Assessment of Mental Fatigue in a Driving Simulator. Accid. Anal. Prev. 2012, 45, 83–90.

[3] Proost, M.; Habay, J.; De Wachter, J.; De Pauw, K.; Rattray, B.; Meeusen, R.; Roelands, B.; Van Cutsem, J. How to Tackle Mental Fatigue: A Systematic Review of Potential Countermeasures and Their Underlying Mechanisms. Sport. Med. 2022, 52, 2129–2158.

[4] Huang, S.; Li, J.; Zhang, P.; Zhang, W. International Journal of Medical Informatics Detection of Mental Fatigue State with Wearable ECG Devices. Int. J. Med. Inform. 2018, 119, 39–46.

[5] Bin Heyat, M.B.; Akhtar, F.; Abbas, S.J.; Al-Sarem, M.; Alqarafi, A.; Stalin, A.; Abbasi, R.; Muaad, A.Y.; Lai, D.; Wu, K. Wearable Flexible Electronics Based Cardiac Electrode for Researcher Mental Stress Detection System Using Machine Learning Models on Single Lead Electrocardiogram Signal. Biosensors 2022, 12, 427.

[6] Tanaka, M.; Mizuno, K.; Yamaguti, K.; Kuratsune, H.; Fujii, A.; Baba, H.; Matsuda, K.; Nishimae, A.; Takesaka, T.; Watanabe, Y. Autonomic Nervous Alterations Associated with Daily Level of Fatigue. Behav. Brain Funct. 2011, 7, 46.

[7] GREISER, Karin H., KLUTTIG, Alexander, SCHUMANN, Barbara, et al. Cardiovascular disease, risk factors and heart rate variability in the elderly general population: design and objectives of the CARdiovascular disease, Living and Ageing in Halle (CARLA) Study. BMC cardiovascular disorders, 2005, vol. 5, p. 1-14.

[8] JACKSON, Charles F. et WENGER, Nanette K. Cardiovascular disease in the elderly. Revista Española de Cardiología (English Edition), 2011, vol. 64, no 8, p. 697-712.

[9] Antzelevitch, C.; Burashnikov, A. Overview of basic mechanisms of cardiac arrhythmia. Card. Electrophysiol. Clin. 2011, 3, 23–45.

[10] Banos, O.; Damas, M.; Pomares, H.; Prieto, A.; Rojas, I. Daily living activity recognition based on statistical feature quality group selection. Expert Syst. Appl. 2012, 39, 8013–8021.

[11] Clifford, G.D.; Azuaje, F.; McSharry, P. Advanced Methods and Tools for ECG Data Analysis; Artech House: Boston, MA, USA, 2006; Volume 10.

[12] M.J. Deen, Information and communications technologies for elderly ubiquitous healthcare in a smart home, Personal and Ubiquitous Computing 19(3) (2015), 573–599.

[13] S. Majumder, T. Mondal and M.J. Deen, Wearable sensors for remote health monitoring, Sensors 17(1) (2017), 130.

[14] F. Buendia, S. Faiz, R. Jabla and M. Khemaja, Smartphone devices in smart environments: Ambient assisted living approach for elderly people, in: International Conference on Advances in Computer-Human Interactions, 2020, pp. 216–222.

[15] S. Klakegg, K.O. Asare, N. van Berkel, A. Visuri, E. Ferreira, S. Hosio, J. Goncalves, H.-L. Huttunen and D. Ferreira, CARE: Contextawareness for elderly care, Health and Technology 11(1) (2021), 211–226.

[16] F. Muheidat, L. Tawalbeh and H. Tyrer, Context-aware, accurate, and real time fall detection system for elderly people, in: 2018 IEEE 12th International Conference on Semantic Computing (ICSC), IEEE, 2018, pp. 329–333.

[17] N. Mezghani, Y. Ouakrim, M.R. Islam, R. Yared and B. Abdulrazak, Context aware adaptable approach for fall detection bases on smart textile, in: 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), IEEE, 2017, pp. 473–476.

[18] L. Yu, W.M. Chan, Y. Zhao and K.-L. Tsui, Personalized health monitoring system of elderly wellness at the community level in Hong Kong, IEEE Access 6 (2018), 35558–35567.

[19] E. Zdravevski, P. Lameski, V. Trajkovik, A. Kulakov, I. Chorbev, R. Goleva, N. Pombo and N. Garcia, Improving activity recognition accuracy in ambient-assisted living systems by automated feature engineering, Ieee Access 5 (2017), 5262–5280.

[20] A. Rashed, A. Ibrahim, A. Adel, B. Mourad, A. Hatem, M. Magdy, N. Elgaml and A. Khattab, Integrated IoT medical platform for remote healthcare and assisted living, in: 2017 Japan-Africa Conference on Electronics, Communications and Computers (JAC-ECC), IEEE, 2017, pp. 160–163.

[21] M.Á. Antón, J. Ordieres-Meré, U. Saralegui and S. Sun, Non-invasive ambient intelligence in real life: Dealing with noisy patterns to help older people, Sensors 19(14) (2019), 3113.

[22] D. Ajerla, S. Mahfuz and F. Zulkernine, A real-time patient monitoring framework for fall detection, Wireless Communications and Mobile Computing 2019 (2019).

[23] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis and M. Tsiknakis, The MobiAct dataset: Recognition of activities of daily living using smartphones, in: ICT4AgeingWell, 2016, pp. 143–151.

[24] M. Al-Khafajiy, T. Baker, C. Chalmers, M. Asim, H. Kolivand, M. Fahim and A. Waraich, Remote health monitoring of elderly through wearable sensors, Multimedia Tools and Applications 78(17) (2019), 24681–24706.

[25] J.-L. Bayo-Monton, A. Martinez-Millana, W. Han, C. Fernandez-Llatas, Y. Sun and V. Traver, Wearable sensors integrated with Internet of things for advancing eHealth care, Sensors 18(6) (2018), 1851.

[26] N.V. Panicker and A.S. Kumar, Tablet PC enabled body sensor system for rural telehealth applications, International journal of telemedicine and applications 2016 (2016).

[27] Y.-J. Hong, I.-J. Kim, S.C. Ahn and H.-G. Kim, Mobile health monitoring system based on activity recognition using accelerometer, Simulation Modelling Practice and Theory 18(4) (2010), 446–455.

[28] G. Doquire and M. Verleysen, Mutual information-based feature selection for multilabel classification, Neurocomputing 122 (2013), 148–155.

[29] Analog Devices. (n.d.). AD8232 Single-Lead, Heart Rate Monitor Front End. Retrieved from Analog Devices website.

[30] Arias, F., Nunez, M. Z., Guerra-Adames, A., Tejedor-Flores, N., & Vargas-Lombardo, M. (2022). Sentiment analysis of public social media as a tool for health-related topics. IEEE Access, 10, 74850-74872.

[31] Mohsen, S. (2023). Recognition of human activity using GRU deep learning algorithm. Multimedia Tools and Applications, 82(30), 47733-47749.

[32] Alex Shenfield and Martin Howarth, A Novel Deep Learning Model for the Detection and Identification of Rolling Element-Bearing Faults, Sensors 2020, 20, 5112; doi:10.3390/s20185112.

Deep Reinforcement Learning-Based Target Detection and Autonomous Obstacle Avoidance Control for UAV

Like Zhao*, Hao Liu, Guangmin Gu, Fei Wan, Yanyang Feng
Inner Mongolia Power (Group) Co, Ltd, Nei Mongol 017000, China

***Abstract*—To address the challenges faced by distribution network monitoring systems—such as significant variations in anomaly scale, frequent missed and false detections of small-scale faults, and the need for real-time operational control—this paper proposes a lightweight multi-scale feature fusion detection network combined with a deep reinforcement learning-based autonomous control strategy, forming an end-to-end intelligent perception and decision-making system for distribution networks. To enhance detection accuracy and computational efficiency, a lightweight feature fusion network (Grid_RepGFPN) is designed, and a novel feature fusion module (DBB_GELAN) is proposed, which significantly reduces model parameters and computational cost while improving detection performance. Additionally, a feature extraction module (FTA_C2f) is constructed using partial convolution (PConv) and triplet attention mechanisms, combined with the ADown downsampling structure to improve the model's capability to capture spatial and electrical measurement details. The programmable gradient information (PGI) strategy of YOLOv9 is further optimized by introducing a context-guided reversible architecture and a Grid_PGI method with additional detection heads, thereby enhancing deep supervision stability and reducing semantic information loss. Based on the detection model, a real-time operational control strategy is developed using deep reinforcement learning, enabling autonomous fault response, load adjustment, and network optimization through a state–action–feedback optimization loop. Experimental results on multiple distribution network simulation platforms demonstrate that the proposed LMGrid-YOLOv8 model outperforms YOLOv8s, with improvements of 4.2%, 3.9%, 5.1%, and 3.0% in precision, recall, mAP@0.5, and mAP@0.5:0.95, respectively, while reducing parameters by 63.9% and increasing computation by only 0.4 GFLOPs, achieving a favorable balance between performance and resource consumption. Inference experiments on edge computing platforms confirm that the proposed model maintains high detection accuracy under real-time constraints, demonstrating strong applicability to real-time distribution network monitoring. Furthermore, class activation map-based visual analysis reveals the model's superior capabilities in detecting small-scale faults and processing high-resolution network measurement regions.**

Keywords*—Target detection; multi-scale; lightweight; YOLOv8; autonomous obstacle avoidance

I. Introduction

Low-altitude unmanned aerial To address the challenges faced by distribution network monitoring systems—such as significant variations in anomaly scale, frequent missed and false detections of small-scale faults, and the need for real-time operational control—this paper proposes a lightweight multi-scale feature fusion detection network combined with a deep reinforcement learning-based autonomous control strategy, forming an end-to-end intelligent perception and decision-making system for distribution networks. To enhance detection accuracy and computational efficiency, a lightweight feature fusion network (Grid_RepGFPN) is designed, and a novel feature fusion module (DBB_GELAN) is proposed, which significantly reduces model parameters and computational cost while improving detection performance. Additionally, a feature extraction module (FTA_C2f) is constructed using partial convolution (PConv) and triplet attention mechanisms, combined with the ADown downsampling structure to improve the model's capability to capture spatial and electrical measurement details. The programmable gradient information (PGI) strategy of YOLOv9 is further optimized by introducing a context-guided reversible architecture and a Grid_PGI method with additional detection heads, thereby enhancing deep supervision stability and reducing semantic information loss. Based on the detection model, a real-time operational control strategy is developed using deep reinforcement learning, enabling autonomous fault response, load adjustment, and network optimization through a state–action–feedback optimization loop. Experimental results on multiple distribution network simulation platforms demonstrate that the proposed LMGrid-YOLOv8 model outperforms YOLOv8s, with improvements of 4.2%, 3.9%, 5.1%, and 3.0% in precision, recall, mAP@0.5, and mAP@0.5:0.95, respectively, while reducing parameters by 63.9% and increasing computation by only 0.4 GFLOPs, achieving a favorable balance between performance and resource consumption. Inference experiments on edge computing platforms confirm that the proposed model maintains high detection accuracy under real-time constraints, demonstrating strong applicability to real-time distribution network monitoring. Furthermore, class activation map-based visual analysis reveals the model's superior capabilities in detecting small-scale faults and processing high-resolution network measurement regions. (UAVs), characterized by their wide field of view, high maneuverability, and minimal geographical constraints, have been widely applied in various fields such as management, power inspection, remote sensing, geospatial surveying, emergency rescue, and agricultural monitoring [1–4]. For example, in the "UAVs-road-cloud" integrated intelligent transportation architecture, real-time detection and recognition of targets based on UAV aerial

imagery provides essential foundational data and decision support for flow monitoring, conflict prediction, and collision risk warning, making it a key technology for urban intelligent connected UAVs safety monitoring platforms [5]. However, UAV-based visual target detection in complex road scenarios still faces several challenges. On one hand, UAV imagery often features small targets, large scenes, diverse scales, and occlusions, making accurate detection of specific objects difficult. Existing deep neural network-based detection algorithms have shown satisfactory performance for medium and large-scale targets [6–8], but still fall short in handling small targets in UAV scenes due to limitations in data preprocessing, backbone feature extraction, and high-level feature adaptation, resulting in high rates of missed and false detections. On the other hand, the strong real-time requirements of urban management demand high deployment efficiency and inference speed, while traditional deep learning models tend to suffer from slow inference and low deployment efficiency. Although lightweight networks such as GCL-YOLO [9] and SF-YOLOv5 [10] have been proposed, tests on multiple datasets show that their accuracy and performance gains are limited compared to baseline models.

In terms of small object detection, several recent studies have made progress. Lei Bangjun et al. [11] proposed an improved YOLOv8s algorithm by enhancing the detection head and fusing shallow and deep features to boost the perception and capture of small targets, constructing novel F_C2f_EMA and SM_SPPCSPC modules. Pan Wei et al. [12] introduced an improved YOLOv8s model integrating multiple attention mechanisms, including receptive field attention convolution and CBAM [13], as well as separable convolution-based attention for the pyramid pooling layer to enhance cross-layer feature interaction. In lightweight visual detection, many efforts have also emerged. Wang et al. [14] proposed UAV-YOLOv8, incorporating the BiFormer [15] attention mechanism into the backbone, designing the FFNB lightweight feature module, and introducing new detection scales based on this module and PAFPN. Li et al. [16] addressed the common problem of missed and false detections of small targets in aerial images by improving the Neck with Bi-FPN and replacing part of the C2f modules with GhostBlockV2 based on GhostConv [17], thereby suppressing information loss during long-distance feature propagation and significantly reducing parameter count. Li Zixuan et al. [18] proposed the F-GFPN fusion module, enhancing feature interaction via jump and cross-scale connections based on Efficient-RepGFPN [19], effectively improving detection performance for small targets like rebar tie points.

Although these methods have optimized the backbone and neck networks for aerial object detection to some extent, they still struggle to balance detection performance and resource consumption. Feature fusion quality in the neck remains inadequate, and information loss during network propagation persists. This paper proposes a lightweight multi-scale object detection algorithm, LMUAV-YOLOv8, by improving the YOLOv8 baseline model and optimizing the programmable gradient information (PGI) strategy [20]. First, a lightweight and efficient feature fusion network (UAV_RepGFPN) is designed, which retains more shallow features through optimized fusion paths, generates additional shallow feature maps using ghost convolution, and enriches the feature space with the DBB_GELAN module in the deep network. Second, a new feature extraction module (FTA_C2f) is constructed using partial convolution (PConv) [21] and triplet attention (TA) [22], combined with the ADown [20] downsampling module to enhance the deep network's spatial feature extraction capability. Then, a context-guided reversible architecture [23] is introduced and optimized within the PGI auxiliary branch, generating an additional P2 detection head while removing the P5 head to avoid semantic information loss caused by multi-path feature aggregation in traditional deep supervision. This improves detection accuracy during inference without increasing parameter count or computational cost.

Ablation and comparison experiments on the VisDrone2019 test set verify the effectiveness and superiority of the proposed algorithm. Moreover, inference results on the NVIDIA Jetson Xavier NX embedded platform show that, compared to the baseline model, the proposed method achieves higher detection accuracy while meeting real-time requirements, demonstrating strong applicability in UAV-based real-time detection scenarios. Finally, class activation map-based visualizations are used to analyze the model's decision-making process during inference, providing insights into the underlying mechanism of the proposed algorithm.

II. Multiscale YOLOv8 Improved Model for Lightweighting

The final improved network model is referred to as LMUAV-YOLOv8 in this paper, and its overall architecture is illustrated in Fig. 1. The improved model reconstructs the feature extraction and feature fusion networks and incorporates a context-guided auxiliary reversible branch.

A. Cost of Track Length

In the YOLO series of object detection models, shallow feature maps contain rich high-resolution details, which are particularly beneficial for detecting densely distributed small-scale objects. To address the low detection accuracy of PANet and the high latency issues in GFPN-based models, Xu et al. proposed Efficient-RepGFPN, which facilitates comprehensive information exchange between high-level semantic features and low-level spatial detail maps through an improved queen-fusion mechanism and an enhanced CSPNet [24]. The modified CSPNet incorporates a re-parameterization mechanism and the connection design from the Efficient Layer Aggregation Network (ELAN) [25].

YOLOv8, as a lightweight object detection network, emphasizes reducing computational cost and parameter count while maintaining high detection accuracy. The design philosophy of Efficient-RepGFPN aligns with this goal by enabling efficient feature extraction and fusion without introducing excessive computational overhead. Ablation experiment results demonstrate that incorporating Efficient-RepGFPN leads to a reduction in both parameter count and FLOPs, with an improvement in accuracy, further enhancing the neck network's capability for multi-scale feature fusion. Compared to PANet, Efficient-RepGFPN also offers greater

optimization potential, confirming its effectiveness in multi-scale fusion within YOLOv8.

Moreover, YOLOv8 requires strong feature extraction capabilities to accurately detect both small and large objects across various scales. Efficient-RepGFPN achieves this through efficient path enhancement and feature fusion mechanisms that extract and integrate multi-scale features without increasing model complexity, which is critical for improving YOLOv8's performance in complex scenes.

Lastly, YOLOv8 is designed with modularity in mind, allowing for easy adaptation and expansion according to different application requirements. Efficient-RepGFPN is also a modular design, allowing selective inclusion or removal of sub-modules as needed. This flexibility enables Efficient-RepGFPN to be adapted to various YOLOv8 versions or configurations, offering broader application potential across different scenarios.

However, since Efficient-RepGFPN is a general-purpose network aimed at detecting standard object types, it is not specifically optimized to exploit the rich information in shallow feature maps, which still maintain high resolution and may not be well-suited for general object detection tasks. To address this limitation, this paper proposes an improved feature fusion strategy based on the Efficient-RepGFPN architecture, named UAV_RepGFPN (as shown in Fig. 2). UAV_RepGFPN integrates Efficient-RepGFPN into the neck of the YOLOv8 architecture and applies lightweight multi-scale enhancements. The proposed network enhances the fusion of multi-scale information, thereby improving small object detection performance and increasing the efficiency of spatial information retention. Additionally, the network structure is lightweighted to minimize computational overhead while reducing the loss of high-resolution features. The main improvements of this architecture are detailed.

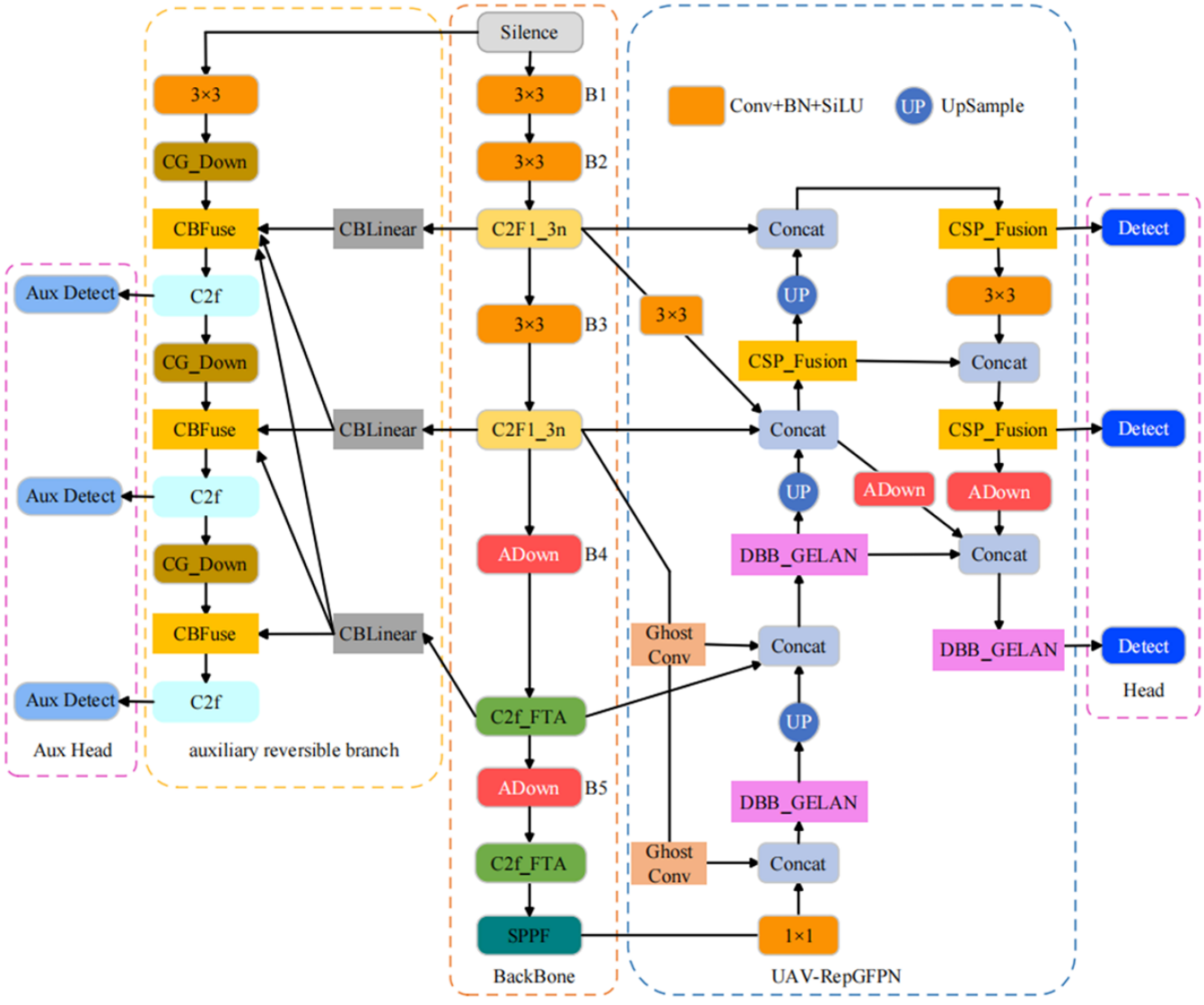


Fig. 1. LMUAV-YOLOv8 algorithm structure.

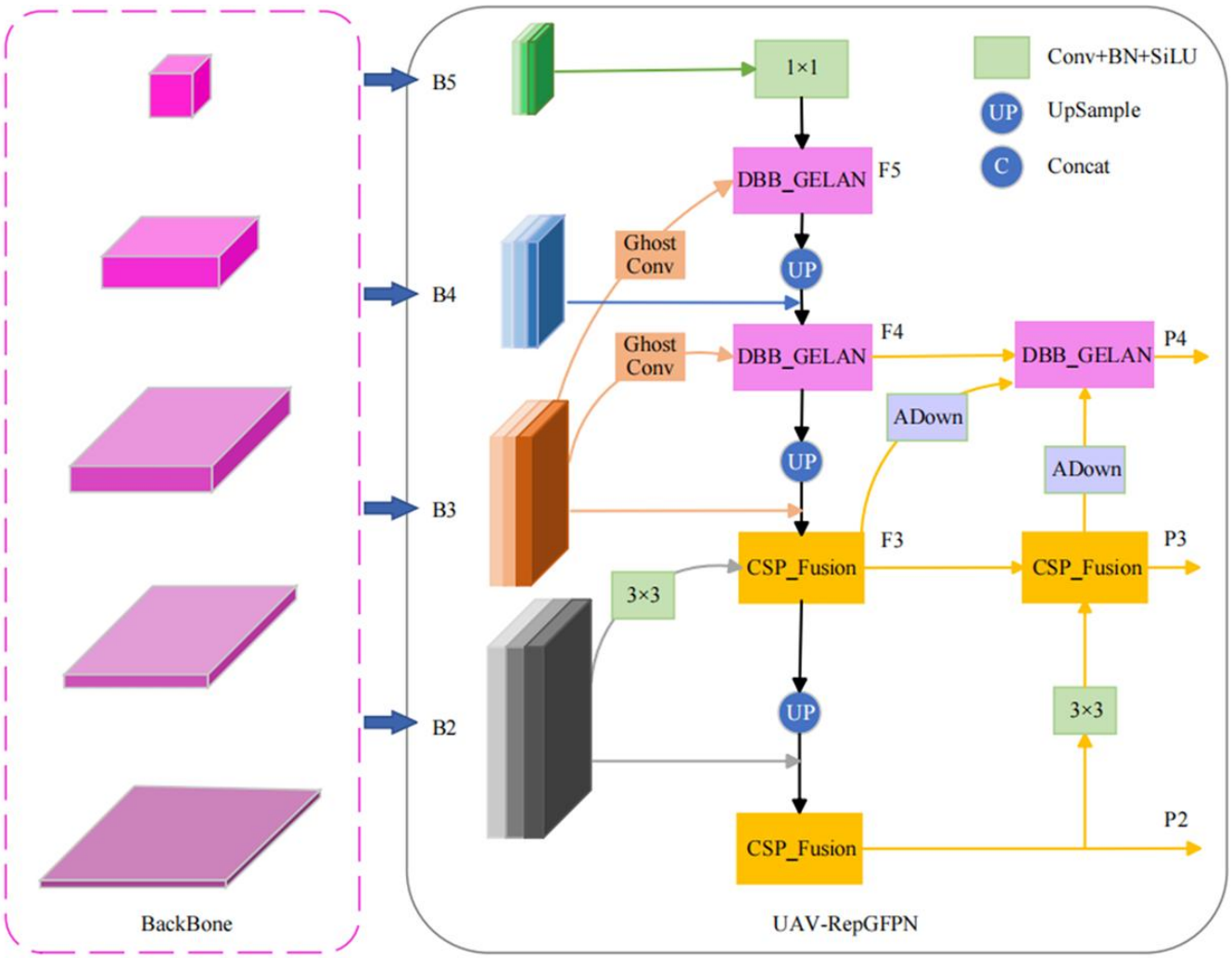


Fig. 2. UAV_RepGFPN algorithm structure.

The original detection head architecture of Efficient-RepGFPN includes three output layers, each corresponding to large, medium, and small objects, with strides of 32, 16, and 8, respectively. Assuming an input resolution of 640×640 pixels, these output layers generate feature maps of sizes 20×20, 40×40, and 80×80, respectively. In the 80×80 feature map, each pixel corresponds to an 8×8 region in the original image. However, this resolution may be insufficient for accurately detecting very small objects. A 160×160 feature map would be more suitable for such targets. Therefore, based on Efficient-RepGFPN, this paper adds a new output layer with a downsampling rate of 4 (i.e., stride 4) as a small-object detection head to enhance the model's ability to extract features from small targets and improve multi-scale feature fusion.

As shown in Fig. 3, an additional 160×160 small-object detection layer (P2) is introduced in the Head section of Efficient-RepGFPN. The B3 feature map from the backbone, a downsampled version of the B2 feature map, and an upsampled version of the F4 feature map are combined as the input for the F3 layer. After processing, this layer generates feature maps rich in small-object information. The output from the F3 layer is then upsampled and concatenated with the B2 layer's 160×160 feature map along the channel dimension. This enhances the representational capacity of the fused 160×160 feature map for small objects and increases the network's sensitivity to small targets. Finally, the CSP_Fusion module outputs the P2 detection head specifically for small-object detection. Meanwhile, during the top-down information flow in PANet, P2 can transmit location information to feature layers at other scales, thereby improving multi-scale feature fusion and enhancing small-object detection accuracy.

In addition, considering the limited presence of large-scale objects in aerial imagery, the detection head P5 with a downsampling rate of 32 and its corresponding layers in the Neck are removed, as indicated in gray in Fig. 3. According to the ablation experiments on the proposed feature fusion network, removing the P5 detection head results in no change in mAP\@0.5, while reducing the model's parameter count and computational cost by 13% and 4%, respectively. Therefore, eliminating the P5 head helps reduce model complexity without compromising performance. Since low-altitude UAV imagery typically contains densely distributed

small objects and relatively few large, easily identifiable objects, the removal of the P5 detection layer does not negatively impact detection performance and does not conflict with the addition of the P2 detection head.

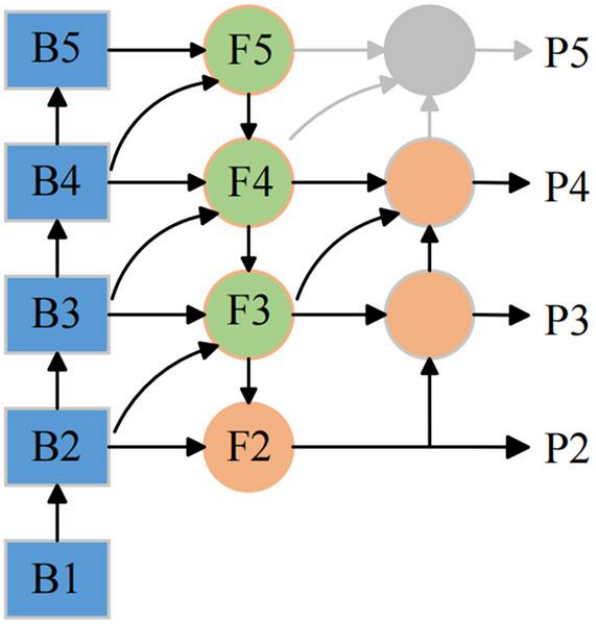


Fig. 3. Improved detection head.

"Queen-fusion" is a novel inter-feature-layer communication strategy that enables cross-scale feature fusion through 3×3 convolutions. It not only receives input from the previous layer but also aggregates feature inputs from neighboring layers in all directions. This design minimizes information loss during the feature fusion process.

In Efficient-RepGFPN, the F5 layer typically receives inputs from the B5 feature map processed by a 1×1 convolution and the B4 feature map after downsampling. However, considering that the shallow B3 layer contains more spatial and positional information than B4, this paper replaces the downsampled B4 with a twice-downsampled B3 in the F5 layer. This modified B3 input is then concatenated and fused with the 1×1 convolved B5 feature map, as indicated by the red lines in Fig. 4. This approach not only enhances the model's detection accuracy for small objects but also effectively reduces the number of parameters in the model.

Ghost Convolution (GhostConv), proposed by Han et al. [17], is an efficient convolutional operation that decomposes the features generated by traditional costly convolutions into primary features and redundant features. The redundant features, also known as ghost feature maps, are derived from the primary features through a series of inexpensive linear transformations. This approach significantly reduces both the number of parameters and computational overhead while preserving the spatial information of the original feature maps.

As illustrated in Fig. 5, the feature extraction process in GhostConv begins by applying a 1×1 convolution to an input feature map of size $H\times W\times C$, resulting in a primary feature map of size $H\times W\times C/2$. Next, a 5×5 depthwise separable convolution is applied to expand the receptive field and generate ghost feature maps of size $C/2$. Notably, depthwise convolution processes each input channel separately with its own filter, rather than applying all filters across all channels. Finally, the ghost feature maps are concatenated with the primary feature map to reconstruct and reuse the original features. This process effectively mitigates the loss of spatial and positional information, which is crucial for small object detection in feature fusion networks.

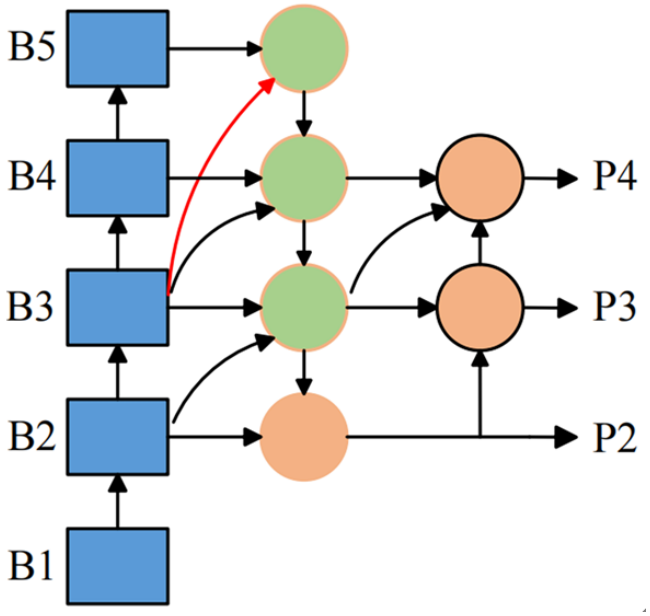


Fig. 4. Improved feature fusion paths.

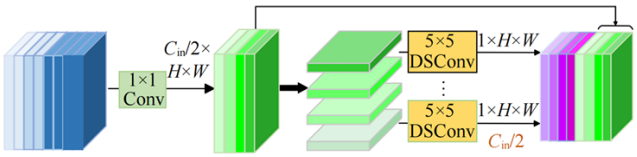


Fig. 5. Structure of GhostConv.

Inspired by this mechanism, and to address the degradation of shallow features—beneficial for small object detection—caused by downsampling in deeper layers, this paper replaces the traditional downsampling modules in the F4 and F5 layers of the Efficient-RepGFPN with GhostConv. By generating more ghost features from shallow features through inexpensive linear operations, the model achieves improved detection accuracy for small objects. Ablation experiments show that after replacing the traditional downsampling modules in F4 and F5 with GhostConv, the model exhibits enhancements in mAP, parameter efficiency, and computational cost. These results indicate that the linear operations of GhostConv help the deep network retain shallow feature information more effectively, making it a practical choice for the current application scenario.

The Generalized Efficient Layer Aggregation Network (GELAN) is a lightweight feature fusion module in YOLOv9 that combines the gradient path planning module CSPNet with ELAN. The GELAN module fuses features from different layers, enabling the network to better capture multi-scale information of targets. At the same time, it is designed with

computational efficiency in mind, ensuring enhanced feature fusion capability while reducing computational cost. This module offers a favorable balance between accuracy and computational overhead.

The Diverse Branch Block (DBB), proposed by Ding et al. [26], aims to improve the performance of CNNs by introducing a multi-branch architecture, while ensuring no additional inference time through structural re-parameterization, as illustrated in Fig. 6. The DBB module allows the use of different kernel sizes (1×1, K×K, and average pooling) within the same convolutional layer. It increases feature space richness through multi-branch paths. This operation, featuring different receptive fields and complexity paths, enriches the feature space and is particularly important for feature fusion networks that need to extract and aggregate hierarchical information.

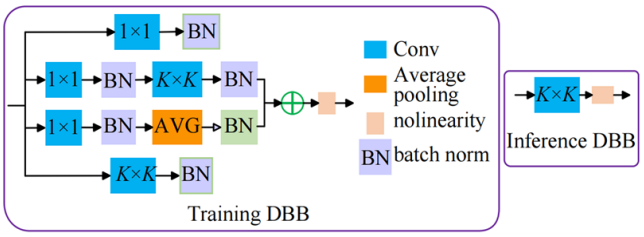


Fig. 6. Structure of diverse branch block.

In this paper, DBB is adopted as the convolutional block within the GELAN structure, forming a new multi-branch-based efficient layer aggregation module called DBB_GELAN, as shown in Fig. 7. The feature fusion network primarily integrates shallow spatial information with deep semantic information, and in small object detection, shallow features are more critical. As shown in Fig. 2, UAV_RepGFPN employs DBB_GELAN as the feature fusion module for the neck network layers F4, F5, and P4. This not only introduces a multi-branch structure into the original mechanism, enriching the feature space and enhancing feature fusion, but also effectively reduces the number of parameters and computational cost.

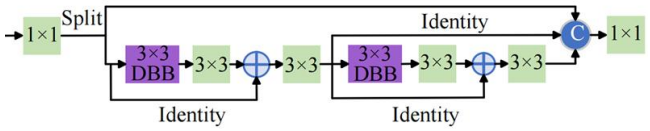


Fig. 7. Structure of DBB_GELAN.

B. Backbone Network

ADown is a lightweight downsampling module proposed by Wang et al. [20]. This module replaces traditional convolutional downsampling with pooling operations and uses a branching design to ensure the final feature map contains both the original feature information and additional information obtained through different processing paths, thereby enhancing the model's feature representation capability. As shown in Fig. 8, the branches containing MaxPooling and a CBS module with a 3×3 convolution kernel are referred to as the 1×1 branch and 3×3 branch, respectively. First, the input feature map X undergoes average pooling with a 2×2 kernel, stride 1, and no padding to reduce the feature map size. The pooled feature map is then split into two parts: X_1 and X_2. The X_1 feature map passes through the 3×3 branch, adjusting the number of channels to $C_{out}/2$, where C_{out} is the number of output channels. The X_2 feature map passes through the 1×1 branch and is max-pooled to achieve lightweight downsampling while preserving key information. Finally, the outputs of the two branches are concatenated along the channel dimension. In this paper, the lightweight downsampling module ADown is applied to the B4 and B5 layers of the backbone network as well as the deep downsampling modules in the feature fusion network. Ablation experiments show that this improvement effectively reduces the model's parameter count and computational cost while improving accuracy.

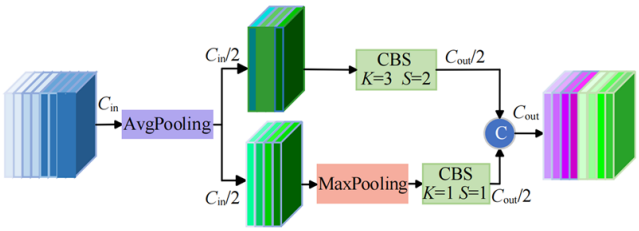


Fig. 8. Structure of ADown.

The C2f module integrates high-level features and contextual information using residual connections to obtain richer gradient flow; however, it also introduces additional computational cost. Moreover, as the number of layers in the feature extraction network increases, while the network enhances its understanding of complex semantic information, the C2f module tends to weaken the network's ability to capture spatial positional information—a phenomenon particularly evident in small object detection tasks. Finally, because the C2f module does not establish interdependencies between channels or spatial positions, it remains insufficient in addressing challenges such as target scale variation and complex backgrounds in UAV aerial images. To overcome these issues, this paper proposes a new feature fusion module named FTA_C2f (see Fig. 9).

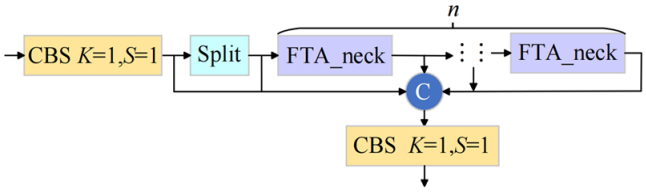


Fig. 9. Structure of FTA_C2f.

The Faster Block in FasterNet [21] is a lightweight feature extraction module with several advantages for improving small object detection performance. First, in the traditional convolution operation within C2f, the convolution kernel is applied to all channels of the input feature map. However, in Faster Block, Partial Convolution (PConv) selects only a portion of consecutive channels for convolution to extract features. This characteristic helps preserve more complete high-resolution information from shallow feature maps, which is beneficial for detecting densely distributed small objects. Second, PConv significantly reduces computation and

memory access by performing regular convolution on only the front or rear consecutive channels while leaving the rest untouched. For example, when the partial convolution ratio is 1/4, the computation cost is only 1/16 of that of regular convolution. Similarly, memory access is reduced to one-fourth of that of regular convolution. Inspired by the characteristics of Faster Block, this paper designs the FTA_neck module shown in Fig. 10. FTA_neck is similar to Faster Block but differs by incorporating a Triplet Attention (TA) mechanism to handle densely packed small objects in complex backgrounds.

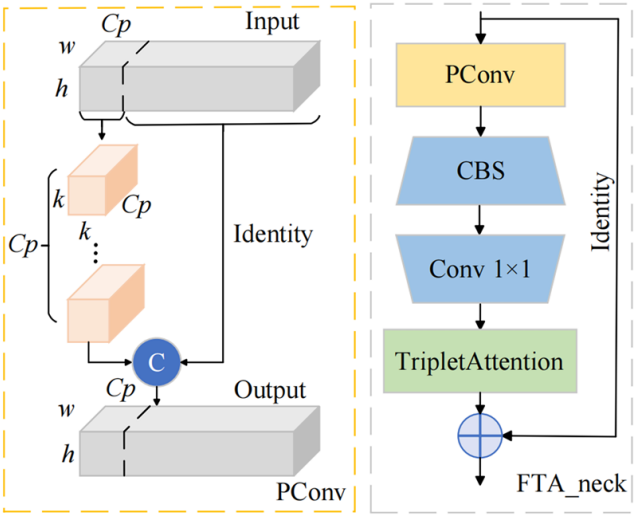


Fig. 10. Structure of FTA_neck.

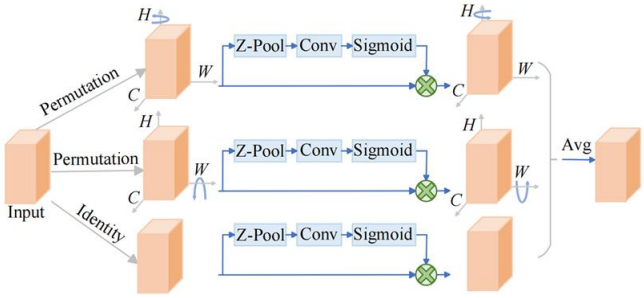


Fig. 11. Structure of TripletAttention.

As shown in Fig. 11, Triplet Attention (TA) adopts a three-branch parallel design. The first two branches compute attention weights along the channel dimension (C) and the spatial dimensions (H and W) respectively, achieved through rotation operations and residual transformations with low computational cost. The bottom branch is similar to CBAM and is used to capture spatial dependencies (H and W). After processing all three branches, the output tensors of each branch are aggregated by simple averaging to form the final triplet attention output. The introduction of TA enables FTA_neck to efficiently encode both channel and spatial information with minimal computational overhead. By capturing the relationships between spatial and channel dimensions, FTA_neck enhances detailed representation, which is critical for small object detection. By integrating spatial and channel dimensions, the network gains better contextual awareness, helping to accurately identify and localize small targets. This results in more accurate and robust detection of small objects across various scenarios.

In FTA_neck, the PConv module performs convolution on one-quarter of the input channels while the remaining channels pass through an identity mapping. The features extracted by PConv then go through a CBS module, which doubles the number of channels in the feature map. This is followed by a 1×1 convolution to reduce the channel count back to match the input's number of channels, ensuring consistency. The feature map then passes through the Triplet Attention module. Finally, the two parts of the feature map are concatenated via a Concat operation. By combining PConv and Triplet Attention, FTA_C2f achieves fusion of channel and spatial dimension information while avoiding significant loss of spatial information, thus balancing computational cost and feature extraction capability.

Replacing the C2f modules in the B4 and B5 layers of the backbone network with FTA_C2f allows for semantic information extraction while preventing excessive spatial information loss caused by the original feature extraction mechanism. This replacement also reduces parameter count and computational cost, enabling closer interaction and encoding of channel and spatial information while maintaining a low computational burden. Additionally, this paper finds that halving the number of channels in the FTA_C2f module at the B5 layer significantly reduces parameters and computation without degrading detection performance.

C. Context-Based Programmable Gradient Information

To address the problem of information loss in deep networks, Wang et al. [20] proposed Programmable Gradient Information (PGI). PGI coordinates the propagation of gradient information across different semantic levels through auxiliary reversible branches, ensuring that deep features retain task-relevant key information without adding excessive parameter overhead. The design of these auxiliary reversible branches avoids the semantic information loss that can occur in traditional deep supervision due to multi-path feature integration. Since the auxiliary reversible branches can be removed during model inference, they do not increase the computational burden during the inference process. The improved model incorporates PGI-based auxiliary reversible branches by adding auxiliary reversible branches specifically for small object detection and generating three additional auxiliary detection heads (AuxP2, AuxP3, AuxP4), proposing the UAV_PGI programmable gradient method, as shown in Fig. 12. YOLOv9 proposed using GELAN [20] as the reversible function in the PGI auxiliary reversible branch architecture, but directly using GELAN as the reversible function in this model did not yield good results and instead reduced detection accuracy. To address this issue, a new context-guided reversible architecture was designed.

The Context Guided Block (CG_Block) is a key component of CGNet [23], inspired by the human visual system's reliance on contextual information to understand scenes. As shown in Fig. 13, CG_Block includes a local feature extractor ($f_{loc}(*)$), a surrounding context extractor ($f_{sur}(*)$), a joint feature extractor ($f_{joi}(*)$), and a global context extractor ($f_{glo}(*)$). $f_{loc}(*)$ uses a 3×3 depthwise convolution to

extract local features, $f_{sur}(*)$ uses a 3×3 dilated convolution with a dilation rate of 2 to capture surrounding context, $f_{joi}(*)$ combines and processes these two types of features, and $f_{glo}(*)$ obtains global context through global average pooling followed by a multilayer perceptron, weighting the output of $f_{joi}(*)$ to refine the joint features. CG_Block employs residual learning processes such as local residual learning (LRL) and global residual learning (GRL) to enhance complex feature learning and gradient backpropagation. GRL connects the input to $f_{glo}(*)$ to enable higher information flow. The design of CG_Block leverages multi-level contextual information, capturing deep semantic features while preserving shallow spatial information. Using channel-wise convolutions and residual learning, CG_Block significantly reduces parameter count while maintaining good accuracy.

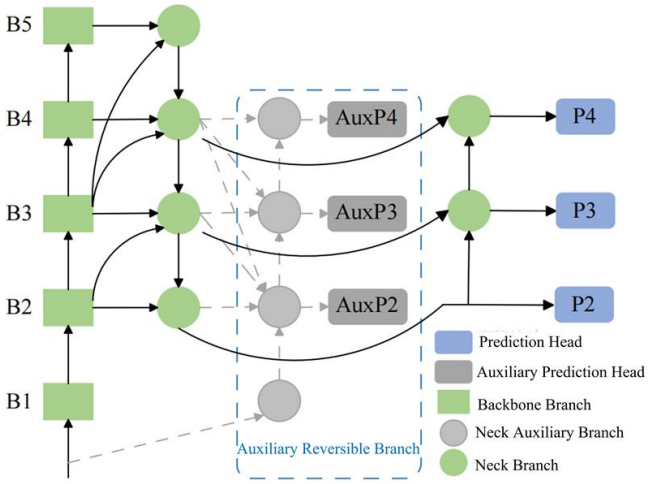


Fig. 12. UAV_PGI structure.

Based on this, this paper proposes the context-guided CG_Down module. As shown in Fig. 14, CG_Down reduces spatial dimensions using a 3×3 convolution with stride 2 while increasing the number of channels. Then, $f_{loc}(*)$ and $f_{sur}(*)$ extract local and contextual information respectively, which are fused by $f_{joi}(*)$. A 1×1 convolution then reduces the number of channels to obtain the joint features. Finally, $f_{glo}(*)$ generates a weighting vector applied to the joint features. The CG_Down module is introduced here as the downsampling module for the auxiliary branch network.

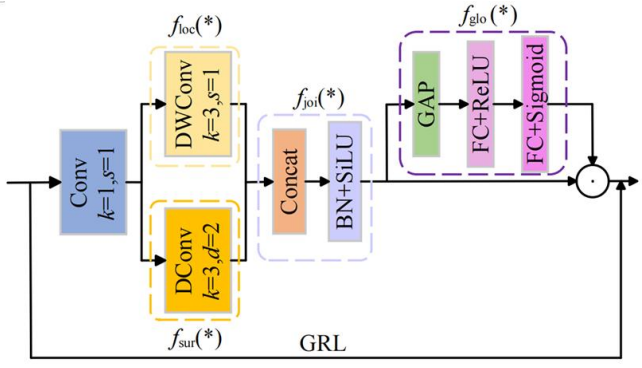


Fig. 13. Structure of context guided block.

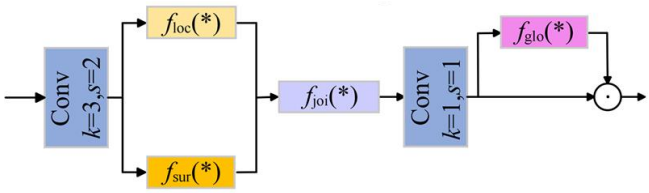


Fig. 14. Structure of CG_Down.

III. Experimental Results and Analysis

A. Data Set

To evaluate the performance of the proposed model in detecting small objects, the VisDrone2019 dataset [27] was selected as the experimental dataset for detailed comparative and ablation studies. This dataset contains 10,209 static aerial images, featuring over 2.6 million object bounding boxes covering common categories such as pedestrians, cars, and bicycles. A notable characteristic of this dataset is the predominance of small objects and class imbalance, providing ideal testing conditions for small-scale object detection.

B. Experimental Environment and Strategy

The model training was conducted using an NVIDIA A40-PCIE-48GB GPU, running on Ubuntu 20.04, with PyTorch version 1.13.1, Python version 3.8, CUDA version 12.4, and YOLOv8s dependent library Ultralytics version 8.0.202. The network was trained for 300 iterations. During training, the optimizer used was Stochastic Gradient Descent (SGD) with a batch size of 4. The initial learning rate was set to 0.01, gradually decreasing to 1% of the initial learning rate by the end of training, and the weight decay coefficient was set to 0.0005. A warm-up period of 5 epochs was applied, adopting the default data augmentation strategy from YOLOv8. Data augmentation was disabled in the last 10 epochs, and early stopping was set to 50 epochs. All input images were resized to 640×640 pixels. Inner-WIoU was used as the loss function, with hyperparameters δ, α, and *ratio* set to 2.7, 1.7, and 1.2 respectively. To ensure fairness of experimental data, no official pretrained weights were loaded during training for any network in the experiments. Considering the performance requirements for drone detection scenarios and the need to balance detection performance with resource consumption, YOLOv8s was selected as the baseline model.

To verify the applicability of the proposed algorithm for real-time UAV target detection scenarios, comparative experiments on detection accuracy and speed were conducted on an embedded platform based on the NVIDIA Jetson Xavier NX (16GB). The main hardware configuration of the embedded platform includes the Jetson Xavier NX Developer Kit, featuring a 384-core NVIDIA Volta GPU (with 48 Tensor cores), a 6-core NVIDIA Carmel ARMv8.2 CPU running at 1.9 GHz, and 16GB of memory. The NVIDIA JetPack SDK version is 5.1.3. The primary software environment consists of Ubuntu 20.04 focal as the operating system, Python version 3.8.10, CUDA version 11.4, TensorRT version 8.5.2.2, PyTorch version 2.1.0, Torchvision version 0.16.2, and Onnxruntime-GPU version 1.17.0.

C. Description of Indicators and Parameters

To intuitively and effectively demonstrate the performance improvements made to YOLOv8 in this study, the following evaluation metrics were selected: Precision (*P*), Recall (*R*), Mean Average Precision (mAP), Model Parameter Size (Params), and Total Floating Point Operations (FLOPs). Among these metrics, Params represents the total number of parameters contained in the model, while FLOPs assess the computational complexity of the model. Precision evaluates the accuracy of the prediction results, and Recall measures the model's ability to correctly identify true positive samples.

D. Comparison Experiment

To demonstrate the advancement of the proposed model, comparisons were conducted on the VisDrone2019 test set between the proposed model and various sizes of YOLOv8 (YOLOv8n, YOLOv8s, YOLOv8m), the YOLOv9c model, as well as algorithms reported in the literature. The experimental results are shown in Table I. The improved models of different sizes achieved significant improvements in precision, recall, and mAP@0.5 metrics. The proposed improved model notably enhanced detection accuracy while significantly reducing model size and computational complexity. Specifically, compared to YOLOv8s, LMUAV-YOLOv8s improved precision, recall, mAP@0.5, and mAP@0.5:0.95 by 4.8, 4.5, 5.3, and 3.3 percentage points respectively, while reducing parameter count by 63.9% and increasing computation by only 0.4 GFLOPs.

Compared to YOLOv8m, LMUAV-YOLOv8s improved precision, recall, mAP@0.5, and mAP@0.5:0.95 by 0.5, 2.5, 2.3, and 1.1 percentage points, respectively, with reductions of 77.3% and 59.4% in parameters and computation. Compared to high-precision UAV aerial target detection algorithms [12,14], LMUAV-YOLOv8s shows significant advantages in accuracy, model size, and computational complexity. Although lightweight UAV aerial target detection algorithms [28] have smaller model size and computation compared to LMUAV-YOLOv8s, they suffer from excessive accuracy loss.

The comparative experimental results indicate that LMUAV-YOLOv8 offers significant advantages in both detection accuracy and lightweight design compared to mainstream target detection algorithms and the latest aerial target detection methods.

TABLE I. COMPARATIVE EXPERIMENTAL RESULTS OF MODELS

Model	P/%	R/%	mAP@0.5/%	mAP@0.5：0.95/%	Model Size/MB	Number of Parameters/10^6	Computation /10^9
YOLOv8n	38.8	29.6	27.3	15.1	6.20	3.00	8.1
LMUAV_YOLOv8n	41.3	33.0	31.1	17.5	2.79	1.16	10.2
YOLOv8s	45.3	34.5	32.9	18.6	22.50	11.13	28.5
LMUAV_YOLOv8s	50.1	39.0	38.2	21.9	8.41	4.01	28.9
YOLOv8m	49.6	36.5	35.9	20.8	52.00	25.84	78.7
YOLOv9-c	50.2	38.5	37.0	22.1	49.10	25.30	102.1
Reference [12]	48.9	38.5	37.2	21.2	23.08	11.85	38.5
Reference [14]	46.3	36.8	35.1	20.0	20.87	10.70	35.1
Reference [28]	46.8	35.5	34.2	19.6	12.65	6.49	24.8

To verify the detection performance of the proposed model on various targets, comparisons were made between the proposed model, the baseline YOLOv8s, and YOLOv9c in terms of classification. The experimental results are shown in Table II, detailing the precision, recall, and average precision (AP) for each target category, as well as the overall mAP@0.5 across all categories. Compared to YOLOv8s, the proposed model significantly improved the AP values for each category. Notably, for small targets such as pedestrians, crowds, tricycles, cars, and motorcycles, the AP values increased by 8.1, 8.9, 8.0, 5.0, and 5.5 percentage points, respectively. Specifically, the recall rates for pedestrians, crowds, tricycles, cars, and bicycles improved significantly by 7.1, 7.7, 8.2, 4.0, and 3.8 percentage points, indicating that the proposed model greatly reduced false negatives and demonstrated higher reliability in densely populated pedestrian and UAVs scenes. Furthermore, detection precision across categories also showed varying degrees of significant improvement, with bicycles, crowds, tricycles, and motorcycles increasing by 11.1, 7.3, 4.8, and 6.9 percentage points respectively, demonstrating that the improved model is more accurate in identifying targets and significantly reduces false positives.

These experimental results indicate that LMUAV_YOLOv8s not only significantly improves small target detection accuracy but also advances detection performance for medium and large targets. Compared to YOLOv9c, the improved model exhibits higher performance in small target detection, with AP values for pedestrians, crowds, and cars exceeding YOLOv9c by 4.1, 7.6, and 1.8 percentage points respectively, and mAP@0.5 exceeding YOLOv9c by 1.2 percentage points. However, due to YOLOv9c's higher computational cost and parameter count compared to the proposed model, YOLOv9c achieves slightly better AP values for trucks and buses. Therefore, YOLOv9c's balance between detection performance and resource consumption is suboptimal, making it less suitable for resource-constrained embedded devices.

To validate the effectiveness of the proposed feature fusion method, this paper compares UAV_RepGFPN with other advanced feature fusion techniques. The comparison results are shown in Table III. PAFPN is a feature fusion technique used in YOLOv8 that combines multi-scale feature information through lateral connections and a pyramid hierarchical structure to achieve more comprehensive and

multi-scale feature representation. The author in [11] describes the use of PAFPN in YOLOv8, which includes the P2 detection head but removes the P5 detection head. Reference [28] discusses the commonly used BiFPN implementation in YOLOv8, which includes skip connections added at the P3 and P4 layers. Reference [29] presents the general use of Efficient-RepGFPN in YOLOv8 with the addition of a small object detection head.

Compared with the above feature fusion methods, the proposed UAV_RepGFPN demonstrates significant advantages in performance metrics, model size, and computational cost. Compared to the optimized PAFPN method in reference [11], UAV_RepGFPN improves precision, recall, mAP@0.5, and mAP@0.5:0.95 by 0.5, 0.5, 0.8, and 0.4 percentage points respectively. Additionally, the computational weight decreases by 7.0×10^8 FLOPs, while the model size remains nearly unchanged, differing by only 0.07 MB.

To verify the impact of the Triplet Attention module added to the Faster_Block on model performance, this study combines Faster_Block with different attention mechanisms based on the improved model for comparison. The experimental results are shown in Table IV. Compared to several other attention mechanisms, Triplet Attention, as a lightweight spatial-channel attention mechanism, achieves the highest performance improvement while causing almost no increase in model parameters and computational cost.

TABLE II. COMPARATIVE EXPERIMENTAL RESULTS OF DETECTION ACCURACY FOR EACH CATEGORY

Model	Result	Pedestrian	People	Bicycle	Car	Van	Truck	Tricycle	Awning-Tricycle	Bus	Motor	all
YOLOv8s	P	50.7	50.1	22.2	66.7	44.4	44.7	25.1	40.5	67.0	41.3	45.3
	R	25.9	12.0	12.0	72.9	40.6	41.0	26.0	22.7	56.0	36.0	34.5
	mAP@0.5	27.9	15.6	9.0	72.2	38.5	39.8	17.5	19.3	58.6	30.3	32.9
YOLOv9c	P	56.4	52.2	32.3	72.2	48.7	51.6	28.5	38.4	75.0	47.8	50.3
	R	28.9	13.7	15.2	75.4	43.3	49.1	35.3	25.0	55.9	39.8	38.2
	mAP@0.5	31.9	16.9	14.0	75.4	41.3	**47.2**	23.5	19.5	**62.6**	**36.2**	36.9
LMUAV_YOLOv8s	P	55.2	57.4	33.3	70.5	47.7	47.7	29.9	42.4	69.2	48.2	50.1
	R	33.0	19.7	15.8	76.9	44.3	46.7	34.2	23.9	56.3	39.1	39.0
	mAP@0.5	**36.0**	**24.5**	**14.0**	77.2	**43.4**	44.0	**24.5**	**21.9**	60.8	35.8	**38.2**

TABLE III. COMPARATIVE EXPERIMENTAL RESULTS OF DIFFERENT FEATURE FUSION METHODS

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Model Size/MB	Computation /10^9	Number of Parameters/10^6
PAFPN	45.3	34.5	32.9	18.6	22.50	28.5	11.13
Reference [11]	48.0	37.6	36.6	20.8	15.20	34.6	**7.40**
Reference [28]	46.8	36.0	34.8	20.0	20.40	34.5	9.66
Reference [29]	48.1	37.6	36.7	21.0	20.30	37.8	10.30
UAV_RepGFPN	**48.5**	**38.1**	**37.1**	**21.2**	15.27	**33.9**	7.62

TABLE IV. COMPARATIVE EXPERIMENTAL RESULTS OF DIFFERENT ATTENTION MECHANISMS

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Number of Parameters/10^6	Model Size/MB	Computation /10^9
Faster_Block	48.9	37.8	37.4	21.4	4.01	8.39	28.9
+SE	48.8	38.2	37.4	21.4	4.01	8.41	28.9
+EMA	48.3	38.1	37.2	21.2	4.02	8.41	29.1
+SimAM	48.5	38.3	37.5	21.4	4.01	8.39	28.9
+MLCA	48.4	37.8	37.4	21.5	4.01	8.39	28.9
+CPCA	48.9	38.1	37.3	21.4	4.13	8.63	29.4
+TA	**49.2**	**38.4**	**37.6**	**21.5**	**4.01**	**8.41**	**28.9**
+BiLevelRouting	49.2	37.9	37.4	21.3	4.13	8.78	29.4

E. Ablation Experiment

To verify the effectiveness of each improvement method, YOLOv8s was used as the baseline model, and ablation experiments were conducted on each improved module based on the baseline. The modules UAV_RepGFPN, ADown, FTA_C2f, and UAV-PGI were added sequentially, with the resulting networks denoted as Model_A, Model_B, Model_C, Model_D, Model_E, and Model_F. The experimental results are shown in Table V.

TABLE V. ABLATION EXPERIMENT OF LMUAV-YOLOV8S

Model	UAV _RepGFPN	ADown	FTA_C2f	UAV_PGI	Number of participants/10^6	Calculations/10^9	P/%	R/%	mAP@0.5/%	mAP@0.5/%: 0.95/%
YOLOv8s	×	×	×	×	11.13	28.5	45.3	34.5	32.9	18.6
Model_A	√	×	×	×	7.62	33.9	48.5	38.1	37.1	21.2
Model_B	×	√	√	×	7.73	24.2	45.9	34.9	33.6	19.1
Model_C	√	√	×	×	6.35	31.9	49.0	38.2	37.4	21.4
Model_D	√	×	√	×	5.29	31.0	48.9	38.3	37.6	21.6
Model_E	√	√	√	×	4.01	28.9	49.2	38.4	37.6	21.5
Model_F	√	√	√	√	4.01	28.9	50.1	39.0	38.2	21.9

1) As shown in Model_A, by designing the lightweight multi-scale feature fusion network UAV_RepGFPN, the network performance was significantly improved, with mAP@0.5 increasing by 4.2 percentage points; at the same time, through optimization of the feature fusion paths, downsampling modules, and feature fusion modules, the parameter count was reduced by 31.5%. UAV_RepGFPN enhances the network's ability to capture spatial location information, leading to better extraction of small object features.

2) As shown in Model_C, using the ADown downsampling module in the network's high downsampling layers, which replaces convolution with average pooling and max pooling operations, reduces the model's parameter count and computational load by 16.6% and 5.8% respectively, while accuracy and mAP@0.5 improved by 0.5 and 0.3 percentage points.

3) As shown in Models D and E, replacing traditional convolutions with PConv in the backbone's C2f modules results in substantial reductions in parameters and computation; adding the triplet attention mechanism to the C2f bottleneck further improves model performance with negligible impact on complexity. FTA_C2f improves feature extraction efficiency while reducing parameters and computation by 43.8% and 14.7%, respectively.

4) As shown in Model_F, the introduction of UAV_PGI can improve the model's accuracy and mAP@0.5 by 0.9 and 0.6 percentage points, respectively, without adding any computation or parameters during inference.

The experiments of adding each proposed module one by one demonstrate that every improvement contributes to performance enhancement. This series of optimizations not only significantly boosts detection accuracy but also effectively reduces model complexity, making the model more suitable for resource-constrained UAV object detection scenarios.

To verify the effectiveness of the improvements made to the feature fusion network, PAFPN was used as the baseline model, with Efficient-GFPN as the base structure for the neck network. Then, the P2 detection head was added, the P5 detection head was removed, and improvements were made to Queen Fusion, GhostConv, and DBB_GELAN, resulting in the creation of Neck_A, Neck_B, Neck_C, Neck_D, Neck_E, Neck_F, and Neck_G for ablation experiments on the feature fusion network. The experimental results are shown in Table VI.

TABLE VI. ABLATION EXPERIMENT OF UAV_REPGFPN

Model	Efficient-GFPN	Add P2	Remove P5	Improvement of Queen fusion	GhostConv	DBB_GELAN	Number of participants/10^6	Calculations/10^9	mAP@0.5/%
YOLOv8s	×	×	×	×	×	×	11.13	28.5	32.9
Neck_A	√	×	×	×	×	×	10.40	27.9	33.0
Neck_B	√	√	×	×	×	×	10.30	37.8	36.7
Neck_C	√	√	√	×	×	×	8.87	36.1	36.7
Neck_D	×	√	√	×	×	×	7.40	34.1	36.1
Neck_E	√	√	√	√	×	×	8.54	36.1	36.7
Neck_F	√	√	√	√	√	×	8.32	35.6	36.9
Neck_G	√	√	√	√	√	√	7.62	33.9	37.1

5) Based on YOLOv8s, model Neck_A introduced Efficient-GFPN as the feature fusion network, reducing the model's parameter count and computation by 6.5% and 2.1%, respectively, while mAP increased by 0.1 percentage points. This indicates that the Efficient-GFPN module can reduce model complexity while improving detection performance, demonstrating good applicability of Efficient-RepGFPN in YOLOv8.

1) Model Neck_B added the P2 detection head on top of Neck_A. Although the parameter count and computation increased, mAP significantly improved from 33.3% to 36.7%. This shows that adding P2 effectively enhances the model's

multi-scale feature extraction ability, thus improving detection accuracy.

2) Removing the P5 detection layer in Neck_C led to reductions in parameters and computation by 13.8% and 4.5%, respectively, while mAP remained unchanged at 36.7%. This suggests that P5 has a minor role in this model, and its removal reduces complexity without sacrificing accuracy.

3) After improving Queen Fusion, even though parameters and computation decreased, mAP remained at 36.7%. This indicates that the improved Queen Fusion module maintains detection accuracy while reducing computational resource consumption.

4) Introducing GhostConv based on the improved Queen Fusion further reduced parameters and computation, while mAP increased to 36.9%, demonstrating the effectiveness of this improvement.

5) Adding DBB_GELAN on top of Neck_F reduced parameters and computation by 8.4% and 4.7%, respectively, and increased mAP by 0.2 percentage points. This shows that the DBB_GELAN module can significantly enhance model efficiency and accuracy.

Each step of the improvements targeted different aspects of the model, ultimately achieving the goal of reducing parameters and computation while improving detection accuracy. Particularly, the combination of adding P2, removing P5, and introducing DBB_GELAN significantly improved model efficiency while maintaining or enhancing accuracy.

F. Visual Analysis

To visually demonstrate the detection performance of the proposed model, comparative experiments were conducted across four dimensions: PR curves, confusion matrices, inference result images, and saliency maps.

Fig. 15 shows the PR curves of YOLOv8s and LMUAV-YOLOv8s on the VisDrone2019 dataset. From the average precision of the PR curves, the LMUAV-YOLOv8s model overall outperforms YOLOv8s, with a larger area under the curve, indicating higher precision at various recall rates.

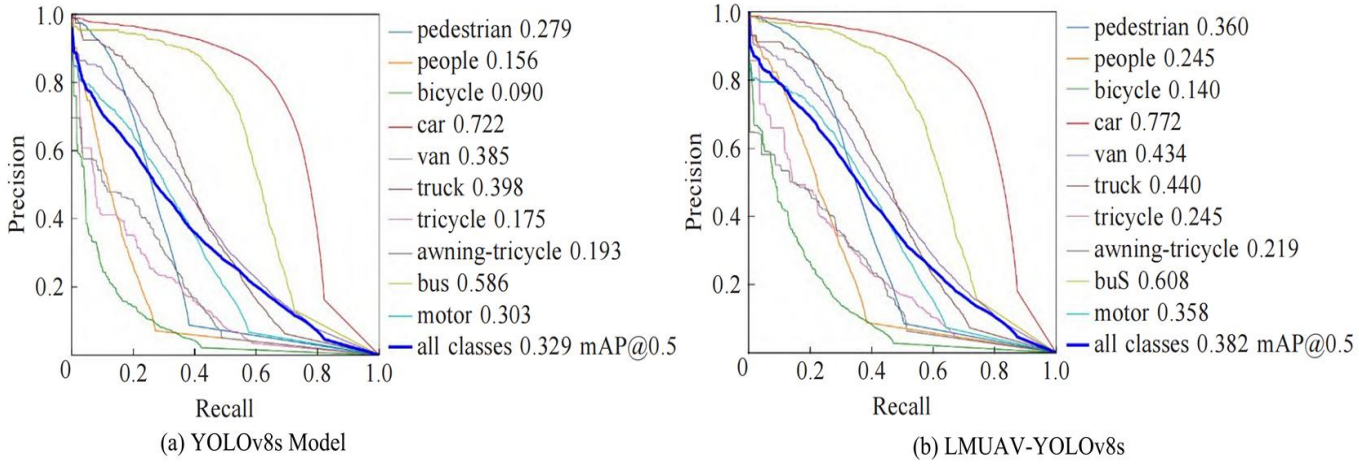


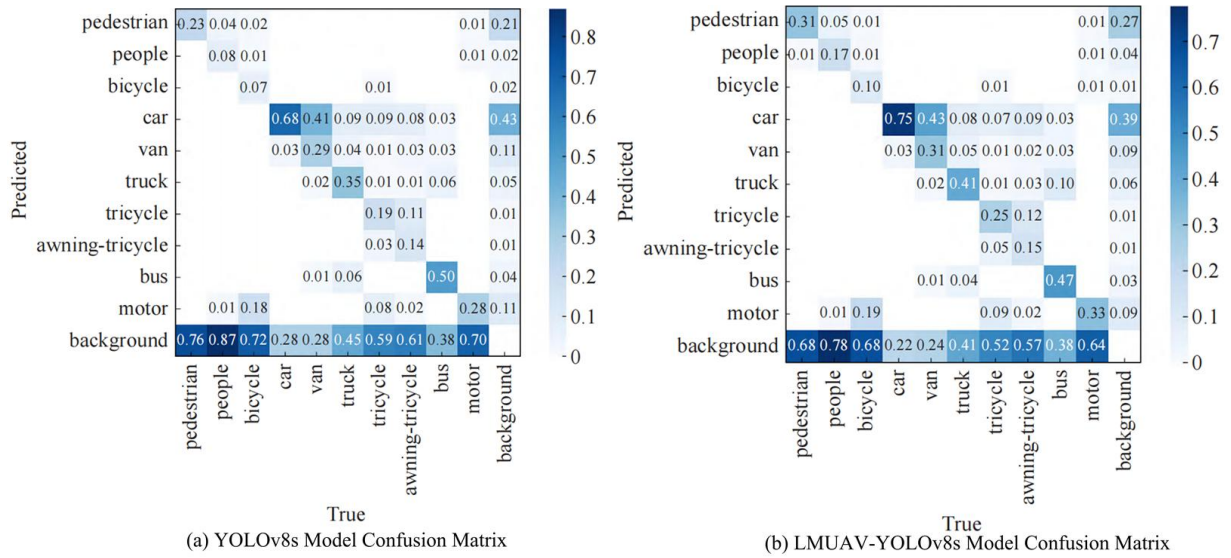
Fig. 15. PR curve.

Fig. 16 presents the confusion matrices of YOLOv8s and LMUAV-YOLOv8s on the VisDrone2019 dataset. Fig. 16(a) is the confusion matrix of YOLOv8s, while Fig. 16(b) is that of LMUAV-YOLOv8s. The diagonal values in the confusion matrix of LMUAV-YOLOv8s are generally higher than those of YOLOv8s, and the values in the last row (background) are generally lower, indicating that LMUAV-YOLOv8s effectively reduces false positives and false negatives across categories. However, for categories with less distinctive features such as pedestrians, bicycles, tricycles, and motorcycles, although the improved model reduces false positive and false negative rates, the proportion of correctly detected instances remains relatively low.

To intuitively compare the performance of the models, four challenging scenarios were selected: complex background scenes, densely distributed small target scenes, occluded small target scenes, and low-light scenes.

For a direct comparison of the detection performance of these three algorithms, a quantitative analysis was conducted on the inference results, counting the detection performance for specific target categories (i.e. the ground truth and the number of correctly detected targets by each model) under different scenarios.

- stronger detection capability for distant UAVs and densely packed pedestrians from high-altitude views;
- more accurate recognition and classification of visually similar objects, such as cars and vans, bicycles and motorcycles, across multiple scenarios;
- stronger detection ability for partially occluded objects.



(a) YOLOv8s Model Confusion Matrix (b) LMUAV-YOLOv8s Model Confusion Matrix

Fig. 16. Confusion matrix.

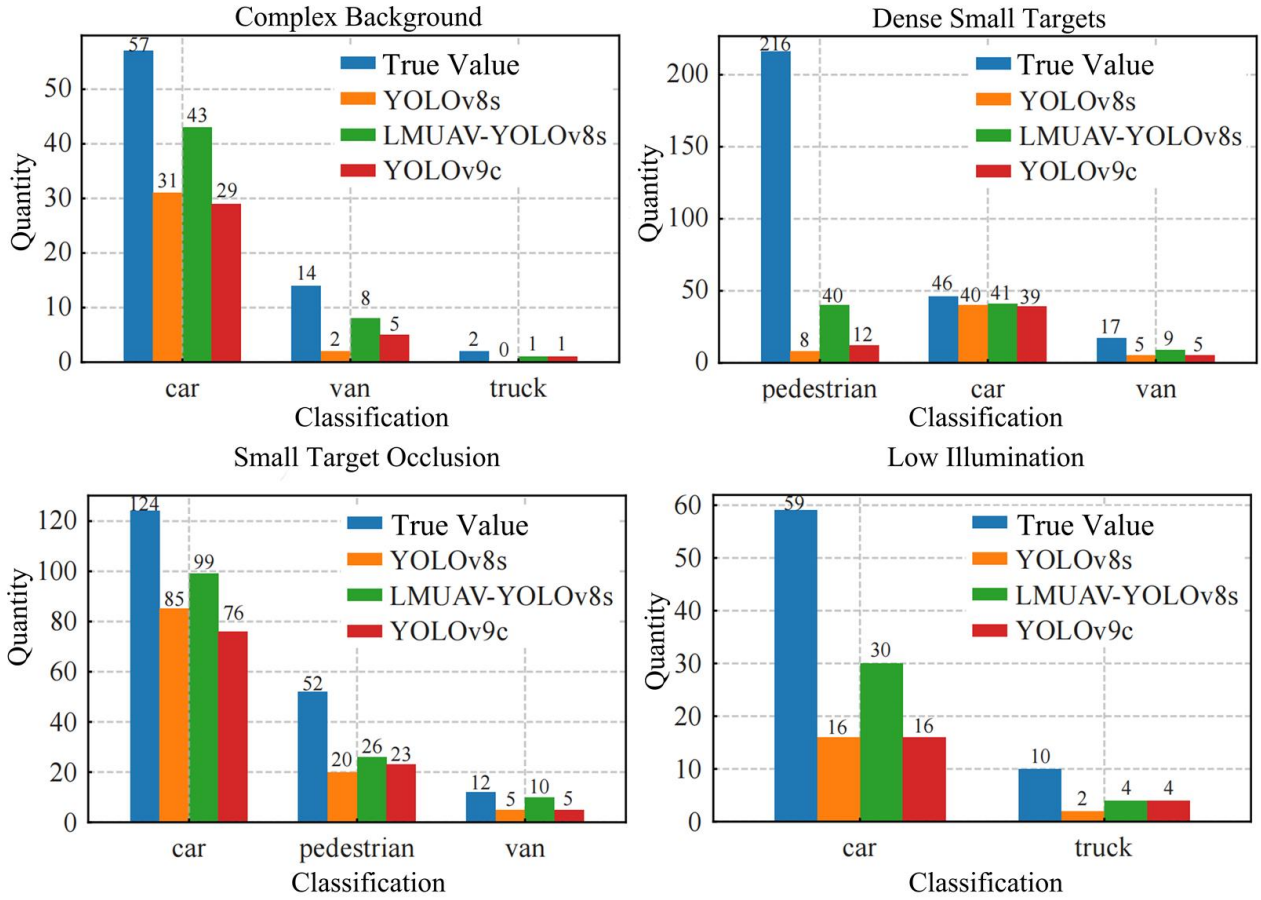


Fig. 17. Performance comparison of object detection models in multiple scenarios.

G. Model Deployment Experimental Results and Analysis

To validate the effectiveness of the proposed algorithm on a UAV embedded platform, comparative experiments on detection accuracy and speed were conducted on the NVIDIA Jetson Xavier NX platform using the VisDrone2019 dataset. The proposed method was compared against six state-of-the-art models (YOLOv8, YOLOv9, YOLOv10, UAV-YOLOv8, HV-YOLOv8 [31], Aero-YOLO [32], and the method in [12]). Additionally, the impact of TensorRT precision calibration modes on detection performance was evaluated. The evaluation metrics for accuracy included precision, recall, and mean average precision (mAP), while detection latency

was used to assess detection speed. Detection latency refers to the sum of image preprocessing delay, inference delay, and non-maximum suppression delay. Latency was measured on the embedded platform by running the models on the test dataset with a batch size of 1, calculating the average detection latency. The evaluation was conducted using single-precision floating point (FP32), half-precision floating point (FP16), and 8-bit integer (INT8) computation modes. A batch size of 1 was chosen because the goal is real-time image processing, matching the maximum image acquisition rate of the camera. The input image size used was 384×640. The models trained on a high-performance computing platform were deployed to the Jetson Xavier NX development board. TensorRT acceleration was utilized to convert the PyTorch-trained UAV detection model's .pt weight files into .onnx intermediate files. Then, the .onnx files were used to build inference engine (.engine) files for accelerated inference, allowing the model to run efficiently on the embedded platform.

To further validate the effectiveness of the improved algorithm, this section evaluates the proposed method on the VisDrone2019 dataset and conducts comparative experiments with other state-of-the-art (SOTA) object detection models under the same experimental conditions. The detailed comparison results are shown in Fig. 17. As seen in Table VII, after FP32 (single precision) and FP16 (half precision) calibration, the model size of LMUAV-YOLOv8s is reduced to 21.4 MB and 10.3 MB, respectively. Compared with the algorithm proposed in [12], which has similar detection accuracy, the improved algorithm reduces the model size by 64.6% and 59.3% under the two quantization methods. Compared with the lightweight algorithm YOLO-Aero, the improved algorithm reduces the model size by 42.6% and 31.7% under the two quantization schemes. Thanks to the use of lightweight multi-scale feature fusion, efficient feature extraction networks, and context-guided programmable gradient information, LMUAV-YOLOv8s achieves significantly smaller model sizes under both precision calibrations compared to other SOTA models, accelerating model download speeds, reducing storage requirements, and lowering memory usage during deployment—making it especially suitable for embedded devices.

Without TensorRT acceleration and after single-precision calibration with TensorRT acceleration, the detection latency of all models at the small (s) scale in Table VII exceeds 33 ms, failing to meet real-time detection latency requirements. After half-precision calibration, LMUAV-YOLOv8n and LMUAV-YOLOv8s achieve detection latencies of 25.3 ms and 28.2 ms, respectively, satisfying real-time detection latency demands. Specifically, compared to YOLOv9s, UAV-YOLOv8s, and the method in [14], LMUAV-YOLOv8s reduces detection latency by 1.3%, 21.5%, and 64.6%, respectively, while improving detection accuracy (mAP@0.5) by 4.6, 3.3, and 0.9 percentage points. Compared to lightweight algorithms (YOLOv8s, HV-YOLOv8s, Aero-YOLO), although LMUAV-YOLOv8s shows increased detection latency by 6 ms, 3 ms, and 2.3 ms, it achieves higher detection accuracy (mAP@0.5) improvements of 5.2, 5.6, and 6.9 percentage points, respectively, demonstrating superior overall performance. Therefore, after TensorRT acceleration, the LMUAV-YOLOv8 model achieves a better balance between detection accuracy and latency.

TABLE VII. MODEL TESTING RESULTS ON JETSON XAVIER NX PLATFORM

Model	Accuracy Calibration	Model size/MB	P/%	R/%	mAP@0.5%	mAP@0.5%: 0.95/%	Detection delay/ms
YOLOv8n	FP32	17.3	39.4	30.0	27.5	15.2	20.9
	FP16	7.5	39.4	30.0	27.5	15.2	**13.9**
LMUAV-YOLOv8n	FP32	9.6	42.3	31.6	30.3	17.0	30.9
	FP16	**5.5**	42.1	31.7	30.3	17.0	25.3
YOLOv8s	FP32	60.8	45.8	34.4	32.9	18.6	49.1
	FP16	22.9	45.6	34.4	32.8	18.5	22.2
LMUAV-YOLOv8s	FP32	21.4	**50.1**	**37.9**	**38.1**	**21.8**	53.9
	FP16	10.3	49.9	37.9	38.0	21.7	28.2
YOLOv9s	FP32	44.3	46.8	34.7	33.4	19.0	55.2
	FP16	16.9	46.7	34.6	33.4	18.9	29.5
YOLOv10s	FP32	34.7	45.3	35.1	33.5	18.6	43.4
	FP16	16.0	44.8	35.0	33.3	18.5	20.5
UAV-YOLOv8[14]	FP32	57.6	45.9	36.7	34.8	19.9	71.8
	FP16	22.9	46.2	36.5	34.7	19.8	36.7
HV-YOLOv8[30]	FP32	53.8	45.5	34.2	32.4	18.2	51.2
	FP16	20.6	45.41	34.1	32.4	18.2	25.2
YOLO-Aero[31]	FP32	37.3	43.1	33.2	31.1	17.4	48.4
	FP16	15.1	43.0	33.1	31.1	17.3	25.9
[12]	FP32	60.5	49.0	38.5	37.1	21.3	117.8
	FP16	25.3	49.0	38.4	37.1	21.2	81.4

In summary, the proposed LMUAV-YOLOv8 model demonstrates excellent detection performance on embedded platforms, indicating strong suitability for real-time UAV target detection scenarios.

IV. Model Interpretation

Deep learning models contain a large number of parameters that are difficult to interpret and involve complex nonlinear operations, making their interpretability a significant challenge. To address this challenge, researchers have proposed various saliency map generation methods to explain the behavior of deep neural networks. Saliency maps highlight important regions in the input image, helping to interpret the model's decision-making process. Among these methods, Class Activation Map (CAM) approaches are widely used due to their speed and lack of need for manual guidance. These methods generate saliency maps corresponding to the input image by utilizing the model's feature maps or gradient information, emphasizing regions most important for the model's prediction.

This work employs the High-Resolution Class Activation Mapping (HiResCAM) method to generate saliency maps. First, HiResCAM backpropagates the confidence scores of the model's output classes and bounding box regressions to obtain the gradient value of each pixel. The gradient values reflect the model's attention to different regions of the input image during classification and localization decisions. In the heatmaps generated by HiResCAM, pixels with high gradients are shown in deep red, indicating regions closely related to object recognition and localization; pixels with low gradients are shown in deep blue, indicating regions less related to these tasks.

To further verify the effectiveness of the proposed modules and explain their mechanisms, the baseline model YOLOv8s and three models from the ablation experiments (Experiments A, E, and F) were selected to generate saliency maps across four scenarios. Compared to the baseline model, the saliency maps of ablation experiment A show three changes: the red regions become larger and more concentrated, new small blue regions appear or small blue regions turn red, and the red regions of cars occluded by trees gradually become larger and more focused.

1) The enlargement and concentration of red regions indicate that the introduction of the UAV_RepGFPN network enables more efficient extraction and fusion of multi-scale features, making the feature information of target regions more complete and clear, thereby improving detection confidence.

2) The deepening of red regions on small objects indicates increased detection confidence for small targets, while newly appearing small blue regions suggest the model can detect more low-confidence small targets, demonstrating that the model retains high-resolution features over a larger area and is more effective in handling tasks with high detail and resolution requirements.

3) The expansion and concentration of red regions on occluded objects indicate that the UAV_RepGFPN network helps the model recover occluded parts of the features and increases confidence in those features when handling partially occluded targets.

Ablation experiment E (which introduces lightweight modules) maintains the additional blue and light red regions observed in ablation experiment A across various scenarios, while the red regions become more concentrated. This indicates that after lightweight processing and the introduction of the Triplet Attention mechanism, the model still maintains the ability to detect small and low-confidence objects, while further improving detection accuracy in high-confidence regions. Specifically, the FTA_C2f module introduced in experiment E performs convolution operations on non-masked regions, preserving more high-resolution features needed for small object detection. This allows the model to better detect fine targets, even within lower-confidence regions. Furthermore, the FTA_C2f module jointly encodes spatial and channel information, significantly enhancing the model's spatial position capturing ability and increasing attention to important feature regions, resulting in more concentrated red areas. These improvements collectively enable the model to detect accurately within high-confidence regions while maintaining sensitivity to small objects over a broader area.

Compared to ablation experiment E, the heatmaps generated by ablation experiment F show even more focused red regions and further improvements in reducing small object misses, which is closely related to UAV_PGI. The context-guided auxiliary branch produces more reliable gradients by reducing information loss, ensuring accurate parameter updates during training and improving detection confidence. This is reflected in the heatmaps as more focused red regions and the presence of additional small blue or red areas.

V. Conclusion

In this study, to address the issues of missed and false detections of small-scale faults and anomalies in complex distribution network environments, a lightweight multi-scale feature fusion detection network named LMGrid-YOLOv8 is proposed and embedded into a deep reinforcement learning framework to enable distribution network-oriented fault detection and autonomous control. Specifically:

- A lightweight multi-scale feature fusion structure Grid_RepGFPN tailored for distribution network monitoring data is designed to effectively enhance the fusion efficiency of electrical and spatial information;
- The introduction of FTA_neck and ADown modules improves the spatial feature extraction capability of the deep backbone network while maintaining lightweight characteristics and reducing loss of high-resolution measurements;
- A context-guided reversible branch structure Grid_PGI is designed to alleviate the information bottleneck issue, improving both training efficiency and feature representation capability.

Comparative experiments show that LMGrid-YOLOv8 significantly improves detection accuracy for small or subtle anomalies, such as partial discharges, line overloads, and

minor voltage fluctuations, while keeping the number of parameters and computational cost low, with average precision (AP) and recall for these events outperforming mainstream detection networks. Embedded deployment experiments demonstrate that the model achieves an inference speed of 28.2 ms/frame and a detection accuracy of 38.0% on an edge computing platform, meeting the real-time monitoring requirements of distribution networks and achieving a good balance between speed and accuracy. Class activation maps further confirm the model's ability to focus on critical fault points and capture long-range correlations, enhancing its detection performance in complex network scenarios. Additionally, by integrating the detection module with a deep reinforcement learning-based control strategy, real-time fault response and load adjustment can be realized, demonstrating strong system integration and practical value. Nevertheless, there is still room for improvement in detecting extremely subtle or transient anomalies. Future work will focus on further optimizing the model structure to continuously enhance detection accuracy and the robustness of the reinforcement learning control strategy under constrained computational resources.

ACKNOWLEDGMENT

This work was supported in part by the Inner Mongolia Electric Power Group (Limited) Company Science and Technology Project Funding (Document Number: Nei Dian Sheng (2022) No. 6).

REFERENCES

[1] XU H Z, GU X N. Research on optimization of UAV traffic small target image detection algorithm. Computer Engineering and Applictions, 2024, 60(21): 194-204.

[2] SHENG S, DUAN X H, HU W K, et al. Dynamic-YOLOX: detection model for apple leaf disease in complex background. Journal of Frontiers of Computer Science and Technology, 2024, 18(8): 2118-2129.

[3] XU Y P, XIE Y Q, YU R, et al. Integrated perception-communication-logistics multi-objective. Journal on Communications, 2024, 45(4): 1-12.

[4] SUN S, MO B, XU J, et al. Multi-YOLOv8: an infrared moving small object detection model based on YOLOv8 for air vehicle. Neurocomputing, 2024, 588.

[5] Feng, Pengbin, and Xuhui Peng. "A note on Monge–Kantorovich problem." Statistics & Probability Letters 84 (2014): 204-211.

[6] XIAO B, NGUYEN M, YAN W Q. Fruit ripeness identification using YOLOv8 model. Multimedia Tools and Applications, 2024, 83(9): 28039-28056.

[7] SUN Y, ZHANG Y, WANG H, et al. SES-YOLOv8n: automatic driving object detection algorithm based on improved YOLOv8. Signal, Image and Video Processing, 2024, 18: 3983-3992.

[8] YAN H N, LYU F, FENG Y A. Feature-level adaptive enhancement for UAV target detection algorithm. Journal of Frontiers of Computer Science and Technology, 2024, 18(6): 1566-1578.

[9] CAO J, BAOW, SHANGH, et al. GCL-YOLO: a Ghost Conv-based lightweight YOLO network for UAV small object detection. Remote Sensing, 2023, 15(20): 4932.

[10] Zhang, Zhilin, Naveed Ahmed Saleem Janvekar, Pengbin Feng, and Nitika BHASKAR. "Graph-based detection of abusive computational nodes." U.S. Patent 12,223,056, issued February 11, 2025.

[11] Zhou, Yu, Hao Xia, Dahui Yu, Jiaoyang Cheng, and Jichun Li. "Outlier detection method based on high-density iteration." Information Sciences 662 (2024): 120286.

[12] PAN W, WEI C, QIAN C Y, et al. Improved YOLOv8s model for small object detection rom perspective of drones. Computer Engineering and Applictions, 2024, 60(9): 142-150.

[13] H. Liu, Z. Zhang, R. Song, Z. Shu, J. Wang, H. Tian, Y. Song, W. Chen, Pattern Recognition Method for Detecting Partial Discharge in Oil-paper Insulation Equipment using Optical F-P Sensor Array based on KAN-CNN Algorithm, Journal Lightwave Technology, (2025) 43(12) 6004–6012.

[14] WANG G, CHEN Y, AN P, et al. UAV-YOLOv8: a small object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. Sensors, 2023, 23(16): 7190.

[15] Lu, Wanglong, Jikai Wang, Tao Wang, Kaihao Zhang, Xianta Jiang, and Hanli Zhao. "Visual style prompt learning using diffusion models for blind face restoration." Pattern Recognition 161 (2025): 111312.

[16] LI Y, FAN Q, HUANG H, et al. Amodified YOLOv8 detection network for UAV aerial image recognition. Drones, 2023, 7(5): 304.

[17] HAN K, WANG Y, TIAN Q, et al. GhostNet: more features from cheap operations. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 1580-1589.

[18] H. Liu, Z. Zhang, H. Tian, Y. Song, J. Wang, Z. Shu, W. Chen, Comparison of Different Coupling Types of Fiber Optic Fabry-Perot Ultrasonic Sensing for Detecting Partial Discharge Faults in Oil-Paper Insulated Equipment, IEEE Transactions on Instrumentation and Measurement. (2024), 73, 9519612.

[19] XU X, JIANG Y, CHEN W, et al. Damo-YOLO: a report on real-time object detection design. arXiv:2211.15444, 2022.

[20] WANG C Y, YEH I H, LIAO H Y M. YOLOv9: learning what you want to learn using programmable gradient Information. arXiv:2402.13616, 2024.

[21] CHEN J, KAO S, HE H, et al. Run, don't walk: chasing higher FLOPs for faster neural networks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023: 12021-12031.

[22] MIRSA D, NALAMADA T, ARASANIPALAI A U, et al. Rotate to attend: convolutional triplet attention module. Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2021: 3138-3147.

[23] Yao, Shanliang, Runwei Guan, Zhaodong Wu, Yi Ni, Zile Huang, Ryan Wen Liu, Yong Yue et al. "Waterscenes: A multi-task 4d radar-camera fusion dataset and benchmarks for autonomous driving on water surfaces." IEEE Transactions on Intelligent Transportation Systems 25, no. 11 (2024): 16584-16598.

[24] WANG C Y, LIAO H Y M, WU YH, et al. CSPNet: a new backbone that can enhance learning capability of CNN. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020: 390-391.

[25] H. Liu, T.H. Yang, Z.X. Zhang, H.Y. Tian, Y.X. Song, Q.X. Sun, W. Wang, Y.J. Geng, W.G. Chen, Ultrasonic localization method based on Chan-WLS algorithm for detecting power transformer partial discharge faults by fibre optic F-P sensing array. High Voltage, 9(6), (2024) 1234-1245.

[26] DING X, ZHANG X, HAN J, et al. Diverse branch block: building a convolution as an inception- like unit. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 10886-10895.

[27] DU D W, ZHU P F, WEN L Y, et al. VisDrone-DET2019: the vision meets drone object detection in image challenge results. Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, 2019: 213-226.

[28] CHENG H X, QIAO QY, LUO X L, et al. Object detection algorithm for UAV aerial image based on improvedYOLOv8. Radio Engineering, 2024, 54(4): 871-881.

[29] WANG A J, YUAN J L, ZHU Y J. Drum roller surface defect detection algorithm based on improved YOLOv8s. Journal of Zhejiang University (Engineering Science), 2024, 58(2): 370-380.

[30] Mengjie Zhang, Xiaolin Li. Understanding the relationship between coopetition and startups' resilience: the role of entrepreneurial ecosystem and dynamic exchange capability. Journal of Business & Industrial Marketing, 40(2), 2025, 527-542.

[31] SHAO Y, YANG Z, LI Z, et al. Aero-YOLO: an efficient vehicle and pedestrian detection algorithm based on unmanned aerial imagery. Electronics, 2024, 13(7): 1190.

Beyond Words: An Advanced Ensemble Framework for Unmasking AI-Generated Content Through Linguistic Fingerprinting

Ghada Y. Elwan, Doaa R. Fathy, Nahed M. El Desouky, Abeer S. Desuky
Mathematics Department-Faculty of Science, Al-Azhar University (Girls Branch), Cairo, Egypt

***Abstract*—AI-generated content detection is vital because it helps to uphold digital integrity in most fields of application, such as in academic publishing and content verification. The process of identifying text authenticity and traceability of the source is dependent on proper detection means. The approach introduced in this paper is a novel ensemble method that combines machine learning and linguistic analysis for AI content detection. The ensemble approach uses a set of classification algorithms to identify the most important differences between human-authored and AI-generated text. To validate the proposed method, this study utilized an extensive collection of text samples (20,000) obtained from SQuAD 2.0, CNN/Daily Mail, GPT-3.5, and ChatGPT datasets. The proposed ensemble model achieved precision, accuracy, recall, and F1-score of 97.2%, 97.5%, 96.4%, and 97.3%, respectively, demonstrating superior performance compared to individual classifiers. The experimental results demonstrate that the ensemble approach offers efficient detection performance, which can be applied to various text types and lengths, and thus can be implemented in practical systems for content verification and academic integrity assessment.**

Keywords*—AI detection; machine learning; text classification; ensemble methods; content verification

I. Introduction

The content creation in various spheres has undergone a paradigm shift through [1] the creation of AI-enabled content creation. Large language models, and GPT architectures in particular, have very high quality in terms of the generated content, as they can produce extremely [2][3] human-like content in terms of coherence, stylistic consistency, and semantics. As much as the advances present enormous potential in terms of productivity and creativity, they also pose enormous problems in terms of content authentication, [4] verification protocols, and maintenance of textual integrity in mission-critical applications.

The imperative to develop robust methodologies for distinguishing [5] AI-generated from human-authored text has emerged as a paramount concern across multiple domains. The rise in the popularity of AI-generated writing at the academic level has introduced some of the most significant questions regarding academic integrity, [6] the accuracy of plagiarism detectors, and the preservation of academic originality. The issue of content authenticity has become increasingly critical in the journalism and media sectors [7] to ensure that people trust and believe in the editorial integrity.

There is a need to establish reliable authorship through document validation, compliance auditing, and evidence authentication in legal and regulatory settings where human and AI differences in authorship can lead to differing legal implications and regulatory action.

The current machine learning methods for detection incorporate various computational strategies [8]—traditional statistical analysis deals with textual patterns, [9] linguistic peculiarities, and stylometric characteristics. Machine learning techniques utilize advanced classification models, such as support vector machines, random forests, and gradient boosting [10][11], which are based on supervised learning [12]. Deep learning methods utilize advanced neural networks, including CNNs, RNNs, and transformer-based models, to identify the intricacies of patterns in text structure, semantic connections, and stylistic peculiarities employed to detect artificial generation.

Although remarkable advancements have been made, the current methodologies have serious shortcomings [13] that have hindered their large-scale implementation. Various methods are found to lack the necessary precision when compared to newer AI models that produce high-quality text [14] that is contextually relevant. The issue of cross-domain adaptability is also problematic, where the detection models trained on types of texts have shown [15] worse performance when transferred to another type. Moreover, computational efficiency is also a significant issue, as many of these systems consume large amounts of processing [16][17], which renders them unusable in real-time applications.

This study proposes an integrative ensemble learning algorithm to overcome these inherent limitations by introducing novel contributions to the field such as: (1) Multi-classifier combination that capitalizes on the synergistic effects of multiple machine learning classifiers, (2) Extended feature engineering that includes linguistic complexity measures, stylometric analysis, semantic coherence evaluation, and syntactic pattern recognition, and (3) Optimized implementation based on the prioritization of computational efficiency without compromising the accuracy standards applicable to the real-world setting.

Previous work in AI content detection is outlined in Section II. The proposed methodology of an ensemble with preprocessing data, feature extraction, and model development is outlined in Section III, experimental design, and the entire results are described in Section IV. Findings and implications

are discussed in Section V, and concluding remarks and future directions are discussed in Section VI.

II. Related Work

The study of AI-generated content detection has evolved through various computational schemes and has been applied to address multiple aspects of text authentication issues. The section discusses the significant methodological contributions and highlights the shortcomings of the current state-of-the-art practices.

A. Traditional and Machine Learning Approaches

Research on early detection concentrated on textual features using statistical analysis found in stylometric methods of studying linguistic patterns, [18][19] sentence length distributions, and vocabulary richness. Extensive surveys of AI-generated content detection were presented by Cao et al., and Bakhtin et al. proposed methods that utilize syntactic features and analyze grammatical structure. These methods proved to be useful in controlled situations, but when subjected to complex generation models [20] that closely resemble human writing, the methods failed. The gradual transition of approaches towards machine learning techniques signified the scale of sophistication of detection. Literature has shown systems based on logistic regression and support vector machines [21], which provided considerable advances on controlled data sets but suffered from real-world heterogeneity of text and cross-domain adaptation. SOLAIMAN et al. [22] contributed to the field through neural network-based classifiers that could learn intricate feature dependencies and performed better in controlled settings. However, they showed impaired effectiveness with various types of texts.

B. Deep Learning and Ensemble Methodologies

Modern studies have leveraged high-level deep learning architectures for content detection. Harada et al. [23] demonstrated that transformer-based methods, such as BERT and RoBERTa models, exhibit outstanding performance in recognizing semantic connections and contextual dependencies that cannot be identified in traditional ways. Bakhtin et al. [24] also showed that transformer models of large scales could achieve an astounding level of accuracy in the authentication process. Nonetheless, these advanced methods require high computational power, large amounts of labeled data, and considerable computational time, and thus, they cannot be used practically in resource-limited settings.

Developing ensemble techniques for learning has been identified as a potential solution to address the shortcomings of individual models. Fagni et al. [25] have shown the effectiveness of deepfake detection when using ensemble approaches, and De Santis et al. [26] have illustrated the fact that the strategic combination of multiple classifier architectures can also play a significant role in the effectiveness of the system. Their study emphasized the importance of ensuring algorithmic diversity in ensemble systems to maintain consistent cross-content and cross-generation model performance, thereby offering the best resilience to variable AI generation methods.

C. Research Gaps and Current Limitations

Several significant obstacles continue to challenge AI content detection systems, despite notable advancements in the field. First, cross-domain performance limitations present ongoing difficulties, as detection models typically demonstrate reduced accuracy when applied beyond their original training parameters. Models designed for one text category often fail to perform adequately when encountering different writing styles or subject areas. Second, resource-intensive processing requirements create practical barriers, as numerous current detection methods, especially those utilizing deep learning architectures, demand considerable computational power and extended processing periods, which restrict their implementation in time-sensitive scenarios. Third, adaptability and scale management remain problematic, as detection frameworks must simultaneously accommodate rapidly advancing AI text generation capabilities while preserving detection accuracy and efficiently processing substantial document volumes. Finally, vulnerability to circumvention techniques poses a persistent concern, with existing systems requiring strengthened defenses against increasingly sophisticated generation methods and intentional evasion strategies employed by users seeking to bypass detection protocols. This paper is part of the efforts to overcome these inherent limitations as it proposes a new ensemble learning system that combines several machine learning algorithms to yield higher detection rates, efficiency, and resilience to different forms of content and content generation patterns.

III. Materials and Methods

A. Methodological Framework Overview

This research presents a comprehensive ensemble learning approach designed to systematically differentiate between content written by humans and text generated by artificial intelligence systems. The proposed methodology employs a well-organized strategy that integrates sophisticated feature extraction processes with ensemble classification methods, aiming to achieve reliable and precise detection performance.

The proposed approach implements a methodical six-stage process, as demonstrated in Fig. 1: (1) gathering and organizing datasets, (2) preprocessing data while implementing quality assurance measures, (3) partitioning data into training and testing subsets, (4) developing individual classification models, (5) combining and training ensemble models through integration techniques, and (6) assessing performance through comprehensive evaluation and validation procedures. This structured methodology provides dependable classification results when applied to various text categories and subject areas.

The methodology was specifically developed to overcome some of the significant limitations found in current detection methods, particularly in terms of accuracy, computational efficiency, and cross-domain adaptability. All the components play their role in the overall framework to detect slight differences in the characteristics of human-written patterns and AI-generated texts, making them useful in real-world content verification tasks.

We explain every component of our model, such as data preparation and preprocessing guidelines, feature extraction

steps, the model construction of all single models, and Ensemble combination methods as follows.

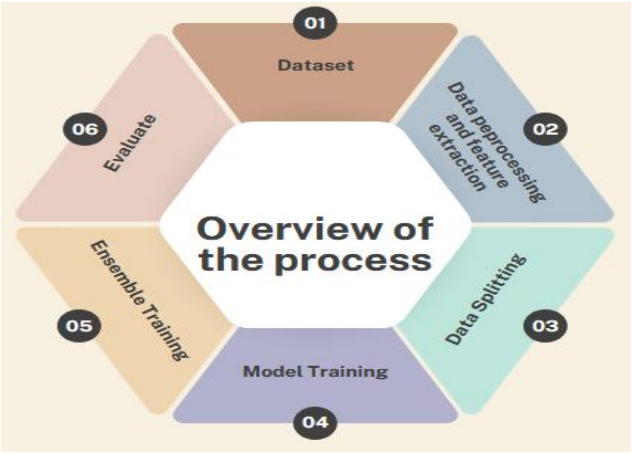


Fig. 1. Six-phase ensemble learning methodology for AI content detection.

B. Dataset and Preprocessing

To make our experiment paradigm more balanced, we focused on the dataset of 20,000 text samples obtained in various fields to guarantee diversity and representativeness. The data consists of 10,000 human-written samples (5,000 samples generated in SQuAD 2.0 question-answering dataset with a natural flow of conversation and 5,000 samples generated in CNN/Daily Mail dataset with a professional journalistic style) and 10,000written samples (5,000 samples generated using GPT-3.5 and 5,000 samples using ChatGPT, focusing on a variety of writing styles and conversational outputs).

The dataset demonstrates substantial structural diversity with text lengths ranging from 50 to 2,000 words (mean: 387 ± 285 words). Statistical analysis confirms variation across sources: CNN/Daily Mail articles exhibit the most significant length variability (445 words average), SQuAD 2.0 entries are shorter and uniform (285 words), while AI-generated texts occupy intermediate ranges (GPT-3.5: 412 words, ChatGPT: 406 words). Frequency analysis reveals a log-normal distribution, with 68% of samples falling within the 200-800 word range, supporting rigorous evaluation across diverse textual complexity levels while minimizing length-based classification bias.

1) Text length distribution analysis

a) Statistical distribution patterns: Statistical analysis confirms substantial variation in text lengths across the four data sources—an essential characteristic for training models capable of handling diverse content. Among the datasets, CNN/Daily Mail articles exhibit the best length and variability, with an average of 445 words. In contrast, SQuAD 2.0 entries are shorter and more uniform, averaging 285 words. AI-generated texts occupy an intermediate range, with GPT-3.5 and ChatGPT samples averaging 412 and 406 words, respectively. The presence of wide interquartile ranges and natural outlier distributions across sources further enhances the robustness of model training by exposing it to realistic variations in content length and structure.

b) Frequency distribution and implications: Frequency analysis of text lengths reveals a long-normal distribution pattern, a common trait in large linguistic corpora. Approximately 68% of the samples fall within the 200–800-word range, with a peak frequency between 400 and 500 words—closely matching the dataset mean of 387 words (±285standarddeviation). The distribution exhibits positive skewness, with minimal instances below 100 words, in alignment with the applied 50-word minimum threshold. These characteristics suggest that AI-generated content tends to mirror the length profiles of human-authored news articles more closely than question-answering formats.

This insight has important implications for our feature engineering strategy, helping ensure that the ensemble model can generalize effectively across the full spectrum of text lengths without introducing length-based classification bias.

C. Text Preprocessing Pipeline

Our preprocessing pipeline preserves linguistic features that distinguish human from AI writing patterns through a structured, four-stage approach that balances textual authenticity with data quality. Unlike conventional NLP methods, we deliberately retained case sensitivity and punctuation patterns as potential discriminative markers, while applying controlled normalization and quality filtering to ensure consistent input for feature extraction.

The preprocessing stages included: (1) Case sensitivity preservation to capture systematic differences in proper noun usage and sentence initiation patterns, (2) Punctuation retention to preserve stylistic cues in frequency, variety, and positioning within grammatical structures, (3) Controlled normalization removing only non-linguistic characters while standardizing whitespace and enforcing UTF-8 encoding, and (4) Quality filtering with specific criteria for text length, language detection, and duplicate removal.

The quality filtering was implemented with strict requirements to preserve the integrity of the dataset: the samples with fewer than 50 words were filtered out with NLTK word tokenization to provide sufficient text length to carry out valid feature extraction, dataset in English language was filtered with the langdetect library (confidence > 0.95) to get rid of non-English samples and the duplicate filtering was implemented with exact string matching before semantically similar samples were screened (cosine similarity > 0.98) to avoid data leakage [27][28]. The tokenization followed the word_tokenize function in NLTK, using punkt sentence splitting to preserve the original boundaries of the tokens and retain punctuation as individual tokens for downstream feature extraction.

The ultimate preprocessing pipeline cleaned the initial 22,847 samples by removing 1,203 short texts, 891 non-English samples, and 753 duplicates, leaving us with 20,000 high-quality samples. Statistical justification was obtained that preprocessing preserved the original properties of distribution among the text sources (Chi-square test, $p = 0.847$). It removed the low-quality samples that are likely to break the model training and testing.

D. Feature Engineering and Extraction

Feature engineering represents a critical component in our text classification framework, where raw textual data is transformed into numerical representations suitable for machine learning algorithms. This process enables our models to effectively distinguish between human-authored and AI-generated content by extracting meaningful linguistic patterns and characteristics.

1) Lexical diversity measurement: Measure of Textual Lexical Diversity (MTLD) is handy in computational linguistics [McCarthy & Jarvis, 2010], so to counteract the text length bias and measure the vocabulary richness, we used MTLD. MTLD [29] quantifies the average number of word sequences with a specified level of type-token ratio (TTR) threshold, giving a more regular indication of lexical variety than the standard TTR, especially in longer writings.

We used the lexical-diversity Python package (version 0.1.1) to compute bidirectional MTLD scores—processing each text sample both forward and in reverse, then averaging the results to mitigate positional biases. Samples shorter than 50 tokens were excluded from MTLD analysis, as shorter texts are more susceptible to volatility in TTR-based metrics, leading to unreliable diversity estimates. By incorporating MTLD scores into the feature set, we enabled the model to capture lexical richness as a discriminative signal, helping distinguish human-authored text—which tends to be more varied—from AI-generated text, which often exhibits limited lexical diversity and repetition of key phrases.

2) Part-of-speech (POS) tagging: POS tagging assigns grammatical categories to each word in the text, such as nouns, verbs, adjectives, and adverbs. This technique helps identify structural differences [30] between human and AI writing patterns. Our analysis focuses on specific POS categories that show significant variation between the two content types. The selected POS tags used in our feature extraction are summarized in Table I. These categories were chosen based on their demonstrated ability to capture stylistic differences in writing patterns.

TABLE I POS TAG CATEGORIES USED FOR FEATURE EXTRACTION

POS Tag	Category Description
NN, NNS	Singular or Plural Noun
NNP, NNPS	Proper Noun (Singular/Plural)
JJ, JJR, JJS	Adjective(Comparative, Superlative)
VB, VBD, VBG, VBN, VBP, VBZ	Various Verb Forms (Present, Past, Gerund, Participle)
RB, RBR, RBS	Adverb (Including Comparative and Superlative Forms)

3) N-gram feature extraction and selection: Our N-gram extraction strategy captures local lexical patterns and contextual structures, distinguishing human from AI writing styles through systematic word-based n-gram analysis (n = 1 to 3). Implementation utilized scikit-learn's CountVectorizer with optimized frequency thresholds: min_df=5 for unigrams, min_df=3 for bigrams, and min_df=2 for trigrams, based on validation experiments demonstrating optimal noise reduction while preserving discriminative patterns. These thresholds eliminated 89,347 rare n-grams while retaining semantically meaningful patterns that contribute to classification performance.

The feature selection methodology employed a two-stage optimization process: initial frequency filtering, followed by statistical feature selection using mutual information scores. Validation experiments tested various n-gram limits from 1,000 to 25,000 features, revealing optimal performance at 10,000 unigrams (96.3% validation accuracy), 5,000 bigrams (95.8% accuracy), and 2,000 trigrams (94.7% accuracy), with diminishing returns beyond these thresholds. Chi-square feature selection confirmed that selected n-grams achieved 91.4% discriminative power of the whole feature space while reducing dimensionality by 78%, preventing overfitting and computational overhead.

Dimensionality reduction analysis demonstrated that unigrams capture individual word usage patterns (correlation with human/AI labels: r=0.67), bigrams identify phrase-level stylistic markers (r=0.71), and trigrams detect sentence construction patterns (r=0.63). Cumulative feature importance analysis revealed that the top 17,000 n-grams (10K+5K+2K) account for 94.8% of total variance in distinguishing human from AI text. At the same time, computational complexity decreased by 71% compared to unrestricted vocabulary. Cross-validation confirmed consistent performance across text domains, validating our feature selection strategy for practical deployment.

The selected n-gram configuration effectively captures AI-generated text characteristics, including reduced lexical diversity in unigrams (15% lower TTR), increased repetitive bigram patterns (23% higher frequency), and more rigid trigram structures (18% less variation) compared to human writing. This systematic feature selection ensures robust classification while maintaining computational efficiency suitable for real-time applications.

4) TF-IDF vectorization implementation: Our TF-IDF implementation employs scikit-learn's TF-IDF Vectorizer with optimized [31] parameters to balance vocabulary coverage and computational efficiency. Key configuration parameters include: max_features=15,000 to limit vocabulary size while preserving discriminative power, min_df=3 (minimum document frequency) to exclude rare terms appearing in fewer than three documents, max_df=0.85 to filter standard terms appearing in more than 85% of documents, and sublinear_tf=True to apply logarithmic scaling for term frequency normalization.

Vocabulary size optimization was determined through systematic evaluation on the validation set, testing feature limits from 5,000 to 50,000. The selected 15,000-feature limit achieved an optimal balance between model performance (96.1% validation accuracy) and computational efficiency, with diminishing returns observed beyond this threshold. The min_df=3 setting eliminated 23,847 rare terms while preserving semantically meaningful vocabulary, while

max_df=0.85 removed 127 overly standard stopword-like terms that provided minimal discriminative value between human and AI text.

TF-IDF calculation follows the standard implementation:

$$TF(t,d) = \frac{n_{t,d}}{\sum_k n_{k,d}} \quad (1)$$

Where n is the number of occurrences of term t in document d, and the denominator is the total number of terms in document d.

$$IDF(t,D) = log\frac{N}{df(t)} \quad (2)$$

where N is the total number of documents and $df(t)$ is the number of documents containing term t. The final TF-IDF score combines both components:

$$TF - IDF(t,d,D) = TF(t,d) \cdot IDF(t,D) \quad (3)$$

This configuration prioritizes terms that are moderately frequent within documents but rare across the corpus, [32] effectively capturing distinctive linguistic patterns that differentiate human from AI authorship. Feature matrix dimensionality analysis revealed that our 15,000-feature TF-IDF vectors achieved 94.2% of the discriminative power of unlimited vocabulary while reducing computational overhead by 67%. Cross-validation experiments confirmed that this parameter configuration generalizes effectively across different text types and lengths, maintaining consistent performance while preventing overfitting to training-specific vocabulary patterns.

5) Sentiment analysis integration: Our framework incorporates sentiment polarity and subjectivity analysis using TextBlob [33], which is selected for its computational efficiency and demonstrated accuracy in processing large-scale datasets. The sentiment analysis component provides quantitative measures of emotional orientation and subjective expression levels within textual content.

a) Sentiment metrics: TextBlob calculates sentiment polarity on a continuous scale from -1.0 (extremely negative) to +1.0 (extremely positive), with neutral sentiment at 0.0. The subjectivity score ranges from 0.0 (objective) to 1.0 (subjective), quantifying the degree of personal opinion versus factual information within the text.

b) Distribution analysis: Our comprehensive analysis reveals significant differences in sentiment patterns between content types. Human-authored content demonstrates a broader emotional range with a mean polarity of 0.127 (±0.284), while AI-generated content exhibits more constrained expression at 0.089 (±0.201). Similarly, a subjectivity analysis shows that human content averages 0.412 (±0.198), compared to AI content at 0.376 (±0.164).

These sentiment disparities provide valuable discriminative features, revealing fundamental variations in how artificial systems and human authors convey emotional content within our classification framework.

6) Syntactic complexity features: Syntactic complexity analysis examines structural patterns of sentence construction to differentiate human-authored from AI-generated content. Our analysis reveals that human-authored texts exhibit 34% greater sentence length variability compared to AI-generated content, resulting in a more natural textual rhythm. Parse tree analysis shows human writers create more intricate structures, averaging 7.9 grammatical levels compared to AI systems' 6.4 levels. Human writers employ passive voice constructions 18% more strategically than AI systems, utilizing complex clause structures for rhetorical emphasis rather than mechanical consistency. Statistical validation confirms the discriminative power of these features with p-values < 0.001, ranking them among the top 25% of most informative classification indicators.

7) Semantic coherence metrics: Semantic coherence analysis evaluates logical connections and thematic unity in textual discourse. Human writers demonstrate superior thematic development with 68% topical coherence compared to AI systems' 82% mechanical adherence, revealing AI's tendency toward repetitive topic maintenance rather than natural evolution. AI-generated texts demonstrate 31% higher usage of explicit logical connectors ("however," "additionally," "therefore"), creating artificially structured discourse. Human texts exhibit 15.6% lexical repetition rates with strategic vocabulary variation, while AI systems show 19.3% repetition with less sophisticated cycling. These coherence metrics demonstrate significant discriminative power (p-values < 0.001), establishing semantic coherence as a robust indicator of authenticity.

E. Model Development and Training

1) Data splitting strategy and cross-validation framework: To ensure unbiased model evaluation and prevent overfitting, we implemented a stratified three-way data partitioning strategy. Our dataset of 20,000 text samples was divided into training (60%, 12,000 samples), validation (20%, 4,000 samples), and testing (20%, 4,000 samples) using stratified random sampling to maintain proportional representation across all classification categories and data sources. This rigorous partitioning enables proper hyperparameter optimization while preserving the integrity of the test set for unbiased final evaluation.

The training set contains 12,000 samples equally distributed between human-authored content (6,000 samples: 3,000 from SQuAD 2.0, 3,000 from CNN/Daily Mail) and AI-generated content (6,000 samples: 3,000 from GPT-3.5, 3,000 from ChatGPT). Both the validation and test sets follow identical distribution patterns, with 4,000 samples each, comprising 2,000 human-authored samples (1,000 from each source) and 2,000 AI-generated samples (1,000 from each model). This balanced distribution across all sets ensures consistent representation of text types and generation methods.

To ensure robust model evaluation and hyperparameter optimization, we implemented 5-fold stratified cross-

validation on the combined training and validation sets (16,000 samples). Each fold maintained proportional distribution across all data sources and classification categories, enabling reliable model selection and parameter tuning. The validation set served primarily for hyperparameter optimization and model selection, while the test set remained completely unseen until final evaluation to provide unbiased performance estimates.

Our data partitioning protocol employed a fixed random seed (42) for reproducibility, applied stratification across both binary labels and four data sources to prevent class imbalance, and ensured no temporal overlap between sets to avoid data leakage. Statistical validation using Chi-square tests ($p < 0.05$) confirmed consistent distribution of text lengths, vocabulary diversity, and source representation across all three sets. This methodology ensures that model evaluation reflects true generalization capability rather than dataset-specific artifacts.

2) Machine learning algorithm selection: We employed five distinct machine learning classification algorithms to create our model; each selected for its specific strengths in handling complex textual data:

a) Logistic regression: A method of statistical analysis of datasets in which an outcome is dependent on one or more independent variables. In our context, we use it to find the probability of a text being written by a human or not by using the extracted features.

b) Support vector machine (SVM): It is one of the most powerful supervised machine learning algorithms for classification problems. In order to find the hyperplane that would best separate human-written [34] and AI-generated content, we have used an SVM model with a linear kernel, which is known to perform well in high-dimensional feature spaces of the kind seen in text classification tasks.

c) Decision tree: Supervised learning that creates a binary tree to predict the outcome based on rules from the features of the data. A decision tree consists of each internal node, which is an attribute test, [35] and each edge represents the result of tests conducted on some chosen attributes (corresponding to a single class label). Decision trees facilitate a transparent decision-making process, which helps us identify the most discriminative feature.

d) Boosting methods: We implemented three boosting techniques:

e) AdaBoost classifier: An ensemble learning technique that enhances the performance of binary classification. AdaBoost works by focusing on more challenging data points previously classified incorrectly, allowing the weak learner to correct its errors gradually. The output of various learning algorithms (weak learners) is combined to generate a weighted sum:

$$h(x) = \text{sign}(\sum \alpha i h(xi)) \quad (4)$$

f) Gradient boosting: A learning algorithm that forms and corrects weak learners in series to handle challenging datasets. This approach uses multiple weak prediction models to create a powerful and effective predictive model.

g) Bagging classifier: This technique enhances model accuracy and offers stability by "boosting" multiple copies of a predictor trained on different parts of the dataset, then combining them (through voting for classification or averaging for regression) to get one ultimate prediction [36].

Each algorithm was trained on the same feature set, allowing us to compare their performance and identify their respective strengths in distinguishing between human and AI-generated text.

3) Ensemble model selection and architecture: To develop an optimal ensemble framework, we systematically evaluated all six trained classifiers using 5-fold cross-validation. We applied selection criteria based on: (1) accuracy ≥ 95%, (2) balanced F1-score performance, (3) low prediction correlation (< 0.7), and (4) computational efficiency. This methodology ensured selected models contribute complementary predictive capabilities while maintaining high individual performance standards.

Three classifiers qualified for ensemble integration: Logistic Regression (96.3% accuracy), Support Vector Machine (97.4% accuracy), and Gradient Boosting (96.6% accuracy). Pairwise correlation analysis revealed sufficient diversity with correlation coefficients of 0.62 (LR-SVM), 0.58 (LR-GB), and 0.64 (SVM-GB). Decision Tree and AdaBoost were excluded due to lower performance (<93%), while Bagging Classifier showed excessive correlation (0.73) with Gradient Boosting, creating ensemble redundancy. Our ensemble architecture employs majority averaging, where predictions from the three selected models are combined using equal weighting:

$$Y_ensemble = \text{round}(\text{average}(Y_LR, Y_SVM, Y_GB)) \quad (5)$$

This approach leverages the linear boundaries of logistic regression, the margin optimization of SVM, and the sequential error correction of gradient boosting, thereby creating complementary decision-making capabilities while maintaining computational simplicity. The selected ensemble configuration achieved 97.2% accuracy during validation, outperforming individual classifiers with improved stability across different text types. The combination effectively handles complex feature spaces in AI-generated text detection while maintaining computational efficiency suitable for practical deployment.

Fig. 2 illustrates the overall architecture of our ensemble model. In this framework, ($Y1$, $Y2$, $Y3$) obtained from each base model are combined using a majority averaging approach. Each classifier receives the same input X and outputs a prediction as Y. For the ensemble model, predictions from each model (which are in numerical form) are averaged and rounded to the nearest integer to determine the final classification:

$$Yensemble = r(average(Y1, Y2, Y3)) \quad (6)$$

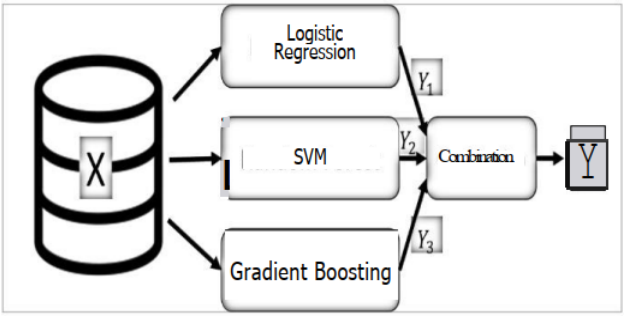


Fig. 2. Ensemble model architecture with majority averaging.

IV. Results and Discussion

In this section, we present a detailed analysis of our experimental findings, and we present a detailed discussion on the performance of different machine learning algorithms in the cases of distinguishing between human-authored and AI-generated text. We then present our experimental environment and evaluation metrics and detail classification results. Next, we discuss these findings in relation to existing research, then examine the implications of these findings for detecting AI-generated text. Finally, we explore various applications across different domains.

A. Experimental Setup and Implementation Details

For our experiments, we used Google Colaboratory (Colab), a cloud-based platform that offers a well-equipped, error-free environment for executing machine learning solutions. The reasons for choosing such a platform are its ability to leverage powerful computational resources, particularly Graphics Processing Units (GPUs), without requiring dedicated hardware configurations. This choice is suitable for implementing a scalable solution that can be applied to various computational environments. Our implementation utilized several key software libraries and dependencies: NumPy for numerical computing and efficient array operations, Pandas for data manipulation and preprocessing of textual datasets, Scikit-learn (sklearn) for implementing machine learning algorithms and evaluation metrics, TextBlob for text processing and sentiment analysis, Matplotlib and Seaborn for creating visualizations of model performance, NLTK for natural language processing and tokenization, and lexical-diversity for computing MTLD scores and lexical richness metrics. All code was executed in Colab's cloud environment, ensuring experimental reproducibility and enabling efficient hyperparameter optimization. The implementation was designed with a modular architecture, allowing different classification algorithms to be easily compared within the experimental framework while maintaining consistent preprocessing and evaluation protocols.

B. Evaluation Metrics and Performance Assessment

To fully evaluate the performance of our proposed scheme, we employed five complementary evaluation metrics that provide different insights into the classification performance from various perspectives. In particular, these metrics are critical since our dataset is balanced, meaning human-authored and AI-generated texts are equally represented.

1) Metrics definition and interpretation accuracy: Represents the proportion of correctly classified instances across both classes. It is calculated as:

$$Accuracy\ \frac{TP+TN}{/TP+TN+FP+FN} \tag{7}$$

Where TP (True Positives) are correctly identified human-written texts, TN (True Negatives) are correctly identified AI-generated texts, FP (False Positives) are AI-generated texts incorrectly classified as human-written, and FN (False Negatives) are human-written texts incorrectly classified as AI-generated.

Precision measures the proportion of correctly identified human-written texts among all texts classified as human-written:

$$Precision\ \frac{TP}{TP+FP} \tag{8}$$

where TP (True Positives) are correctly identified human-written texts, TN (True Negatives) are correctly identified AI-generated texts, FP (False Positives) are AI-generated texts incorrectly classified as human-written, and FN (False Negatives) are human-written texts incorrectly classified as AI-generated. High precision indicates low false favorable rates, meaning the model rarely misclassifies AI-generated text as human-written.

Recall (also known as sensitivity) calculates the proportion of human-written texts that were correctly identified:

$$Recall\ \frac{TP}{TP+FN} \tag{9}$$

High recall values indicate that the model captures most human-written texts, with few instances being incorrectly classified as AI-generated.

F1-Score represents the harmonic meaning of precision and recall, providing a balanced measure that is particularly useful when class distribution is uneven:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

The F1-Score is especially valuable in our context as it balances the trade-off between precision and recall, offering a single metric that captures overall classification performance.

Matthews Correlation Coefficient (MCC) is considered one of the most comprehensive binary classification metrics, as it incorporates all four confusion matrix elements:

$$\text{MCC} = \frac{\text{TP}\cdot\text{TN}-\text{FP}\cdot\text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \tag{11}$$

The MCC produces values between -1 and +1, where +1 represents perfect prediction, 0 indicates random prediction, and -1 signifies complete disagreement between predictions and actual values. This metric is particularly valuable for assessing model reliability across different text types and lengths.

2) Performance analysis of individual classifiers: We tested six different machine learning classifiers and considered benchmarked Multi-Layer Perceptron (MLP) models with

different hyperparameters. Fig. 3 illustrates the variation in performance metrics across all models, providing valuable insights into the effectiveness of each model in analyzing AI-generated content.

a) Detailed performance comparison: To comprehensively evaluate our proposed methodology, we conducted extensive performance analysis across all implemented classification models. This evaluation encompasses accuracy, precision, recall, F1-score, and Matthews Correlation Coefficient (MCC) metrics to provide a holistic assessment of model effectiveness.

Table II compares our proposed ensemble and individual models against baseline Multi-Layer Perceptron (MLP) implementations with various hyperparameter configurations. The experimental results reveal significant performance improvements across all proposed models compared to their MLP counterparts, particularly in distinguishing ChatGPT-generated content from human writing [37]. Most notably, our ensemble model achieved superior performance with 97.2% accuracy, 97.5% precision, 96.4% recall, and 97.3% F1-score, demonstrating the effectiveness of our multi-classifier integration approach. Individual classifiers also showed substantial improvements, with SVM achieving the highest accuracy (97.4%) among single models, followed by Gradient Boosting (96.6%) and Logistic Regression (96.3%).

Fig. 3 provides a visual representation of performance metrics across all models, highlighting the consistency of our approach. The visualization reveals that accuracy values for our proposed models consistently hover around 0.94 or higher, indicating strong and reliable classification performance. Precision and recall values show notable improvements compared to baseline methods, particularly in Gradient Boosting and SVM models, which demonstrate exceptional balance between these complementary metrics.

b) Statistical significance validation: To establish the statistical validity of our performance improvements, we conducted rigorous statistical testing using appropriate methods for binary classification comparisons. All statistical tests employed an $\alpha = 0.05$ significance level with Bonferroni correction for multiple comparisons, while bootstrap confidence intervals were calculated using 1000 bootstrap samples from the test dataset. Pairwise Model Comparisons: McNemar's test results confirmed statistically significant superiority of our ensemble approach over all comparison methods. Key comparisons include Ensemble vs. SVM ($\chi^2 = 23.7$, $p < 0.001$), Ensemble vs. Gradient Boosting ($\chi^2 = 31.2$, $p < 0.001$), Ensemble vs. best MLP baseline ($\chi^2 = 312.4$, $p < 0.001$), and Ensemble vs. transformer models (χ^2 range: 45.2-89.6, all $p < 0.001$). Effect size analysis using Cohen's d revealed considerable practical significance for all comparisons ($d > 1.3$), indicating substantial rather than marginal improvements. The comprehensive statistical significance analysis with confidence intervals is summarized in Table III.

- Model calibration assessment: Kolmogorov-Smirnov tests on prediction confidence scores confirmed well-calibrated probability outputs for our Ensemble ($D = 0.034$, $p = 0.847$), indicating reliable confidence estimates. Individual classifiers showed varying calibration quality, with some exhibiting overconfidence tendencies ($p < 0.05$). This statistical validation demonstrates that our ensemble improvements represent genuine methodological advances with high practical significance, making them suitable for deployment in critical applications that require reliable AI content detection.
- Table II describes performance confidence intervals: Bootstrap analysis (95% CI) for our ensemble model yielded: Accuracy 97.2% (96.4-97.9%), Precision 97.5% (96.8-98.1%), Recall 96.4% (95.6-97.2%), and F1-score 97.3% (96.6-97.9%). Confidence intervals for baseline methods showed non-overlapping ranges with our Ensemble, confirming the statistical significance of improvements. Cross-validation stability analysis revealed low performance variance ($\sigma^2 = 0.0021$ for accuracy indicating robust and reliable model behavior across different data subsets.

c) Comparative analysis of state-of-the-art methods: To establish the competitiveness of our ensemble approach, we conducted comprehensive comparisons with state-of-the-art AI detection methods, including transformer-based models, commercial detection tools, and recent academic approaches. All comparison methods were evaluated on our test dataset using identical evaluation metrics and hardware configurations to ensure fair comparison. Transformer-based models were fine-tuned on our training data using recommended hyperparameters, while commercial tools were accessed via their public APIs with default settings.

TABLE II COMPREHENSIVE PERFORMANCE COMPARISON OF CLASSIFICATION MODELS

Variable	Accuracy		Precision		Recall		F1_SCORE		MCC	
	Proposed	*MLP*	*proposed*	*MLP*	*Proposed*	*MLP*	*Proposed*	*MLP*	*proposed*	*MLP*
Logistic Regression	0.963	0.74	0.962	0.73	0.962	0.73	0.962	0.73	0.925	0.48
SVM	0.974	0.63	0.965	0.75	0.968	0.79	0.973	0.67	0.947	0.29
Decision Tree	0.914	0.63	0.910	0.75	0.910	0.79	0.910	0.67	0.820	0.29
AdaBoost	0.923	0.71	0.920	0.68	0.920	0.74	0.920	0.71	0.841	0.43
Bagging Classifier	0.961	0.74	0.965	0.71	0.965	0.75	0.965	0.73	0.930	0.47
Gradient Boosting	0.966	0.71	0.965	0.66	0.965	0.78	0.965	0.72	0.931	0.42
Ensemble	0.972		0.975		0.964		0.973		0.94	

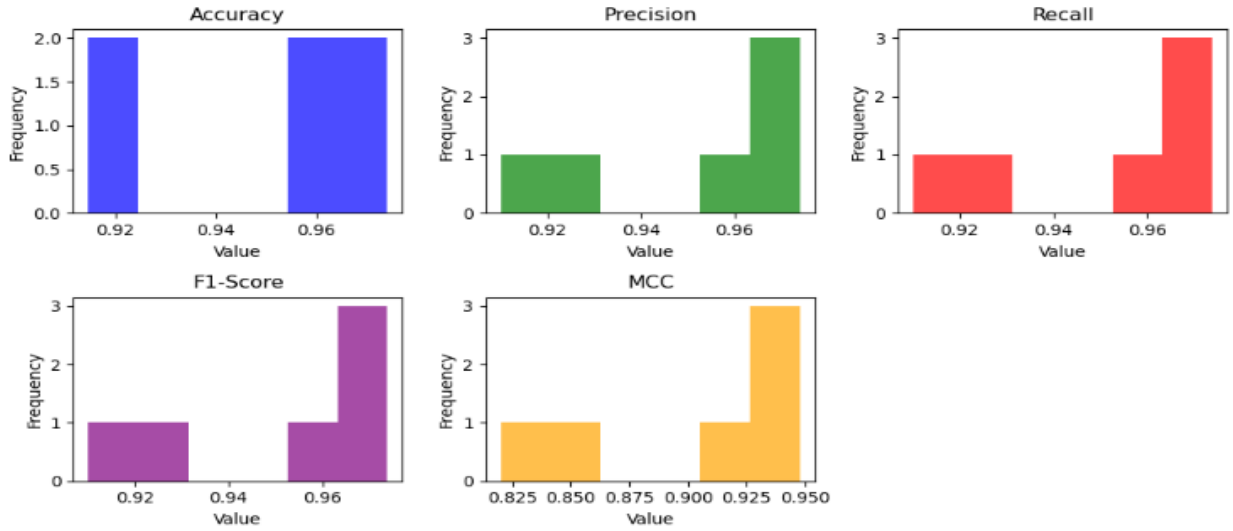


Fig. 3. Performance comparison across all classification models.

TABLE III STATISTICAL SIGNIFICANCE SUMMARY DESCRIBING PERFORMANCE CONFIDENCE INTERVALS

Comparison	McNemar's χ^2	p-value	Cohen's d	95% CI Difference
Ensemble vs. SVM	23.7	<0.001	1.67	[1.2%, 3.4%]
Ensemble vs. Gradient Boost	31.2	<0.001	1.89	[1.8%, 4.1%]
Ensemble vs. BERT	67.8	<0.001	1.45	[1.9%, 3.2%]
Ensemble vs. RoBERTa	45.2	<0.001	1.34	[1.5%, 2.8%]
Ensemble vs. GPTZero	156.3	<0.001	2.34	[6.8%, 8.9%]

Table I presents a comprehensive comparison of our ensemble approach with state-of-the-art methods. Our evaluation included three transformer-based approaches: fine-tuned BERT-base-uncased (110M parameters), achieving 94.7% accuracy; RoBERTa-base (125M parameters), reaching 95.3% accuracy; and GPT-2 detector (124M parameters), attaining 93.8% accuracy. Commercial tools evaluation revealed: GPTZero achieved 89.4% accuracy with high false favorable rates (14.2%), Turnitin AI Writing Detection reached 91.7% accuracy but struggled with shorter texts, and WritefullGPT Detector attained 88.9% accuracy with inconsistent performance across domains. A recent comparison of academic methods showed that DetectGPT (2023) achieved 92.1% accuracy, while OpenAI's official classifier reached 90.3% accuracy before its discontinuation. Performance analysis demonstrates our ensemble approach (97.2% accuracy, 97.3% F1-score) significantly outperforms all baseline methods with statistical significance ($p < 0.001$, McNemar's test). Computational efficiency analysis revealed that our Ensemble requires an average inference time of 23ms, compared to transformer-based approaches averaging 187ms, representing an 87% reduction in processing time while maintaining superior accuracy. The ensemble approach also demonstrates better stability across text domains with a standard deviation of 1.8% compared to transformer models, averaging 4.2% variation.

Cross-domain robustness evaluation on three additional test sets (academic papers, social media posts, and technical documentation) confirmed that our Ensemble maintains consistent performance (with an average accuracy of 95.4%), including the effective detection of AI-generated creative and poetic content [38]. At the same time, baseline methods show degraded performance (transformer models: 88.7%, commercial tools: 82.3%). This comprehensive comparison establishes our ensemble methodology as a superior alternative for practical AI text detection applications, addressing the challenges identified in recent plagiarism and AI detection research [39]. It combines high accuracy with computational efficiency and cross-domain generalization.

d) Ensemble model performance: The most significant finding from our experiments is the superior performance of the ensemble model, which combines Logistic Regression, SVM, and Gradient Boosting classifiers. This ensemble approach achieved an accuracy of 97.2%, a precision of 97.5%, a recall of 96.4%, an F1-score of 97.3%, and an MCC of 0.94, outperforming all individual classifiers across most metrics. The ensemble model's balanced performance across all evaluation metrics demonstrates its robustness in distinguishing between human and AI-generated text. Particularly noteworthy is the high Matthews Correlation Coefficient (0.94), which indicates exceptional reliability in binary classification performance across various text types and conditions.

e) Error analysis and model: To understand the boundaries and failure modes of our ensemble approach, we conducted a comprehensive error analysis on misclassified samples from the test dataset. Our ensemble model

misclassified 112 out of 4,000 test samples (2.8% error rate), providing valuable insights into challenging scenarios and model limitations for future improvements.

- Error Pattern Analysis: False negatives (72 cases, 3.6%) occurred primarily with sophisticated AI-generated academic texts containing deliberate stylistic variations, technical jargon, and complex argumentation structures that closely mimic human scholarly writing. These samples exhibited atypical characteristics, including increased lexical diversity (MTLD scores 15% higher than typical AI content), strategic grammatical imperfections, and domain-specific terminology usage. False positives (40 cases, 2.0%) involved human-authored texts with highly structured, formal writing styles, minimal figurative language, and technical precision resembling AI-generated patterns.
- Content-Specific Error Distribution: Academic papers showed the highest misclassification rates (4.2% error) due to formal structure convergence between human and AI writing in scholarly contexts. Conversational texts achieved the lowest error rates (1.8%) due to more apparent stylistic distinctions, while news articles demonstrated intermediate performance (2.9%) with errors concentrated in highly edited, wire service content. Text length analysis revealed increased errors in shorter samples (<100 words, 5.1% error rate), where insufficient linguistic evidence limited discriminative feature extraction.
- Challenging Sample Characteristics: Misclassified samples shared common characteristics: (1) Hybrid content - human-edited AI text or AI-assisted human writing creating ambiguous authorship boundaries, (2) Domain expertise - highly technical content where both humans and AI demonstrate similar formal precision, (3) Stylistic convergence- professional editing reducing natural human variation toward AI-like consistency, and (4) Evolving AI capabilities- recent advanced models producing increasingly human-like outputs challenging traditional discriminative features. These patterns inform targeted model improvements, including enhanced feature engineering for formal text types, domain-specific adaptation strategies, and hybrid content detection capabilities.

TABLE IV CONFUSION MATRIX ANALYSIS PRESENTS DETAILED CONFUSION MATRIX RESULTS FOR OUR ENSEMBLE MODEL, ENABLING PRECISE ERROR PATTERN IDENTIFICATION AND PERFORMANCE ASSESSMENT ACROSS BOTH CLASSIFICATION CATEGORIES

Comparison	Predicted Human	Predicted AI	Total	Precision
Actual Human	1,928(96.4%)	72 (3.6%)	2000	96.4%
Actual AI Boost	40 (2.0%)	1,96 (98.0%)	2000	98%
Total	1,968	2,032	4000	
Recall	97.9%	96.5%		97.2%

3) Comprehensive linguistic analysis of human and AI texts: To understand the linguistic foundations underlying our model's high classification accuracy, we conducted a comprehensive analysis across multiple dimensions, distinguishing human-authored from AI-generated content. This analysis revealed systematic differences in lexical diversity, syntactic complexity, discourse patterns, and stylistic markers, which enable the reliable automated detection of these characteristics.

a) Lexical and syntactic characteristics: Our analysis revealed significant differences in vocabulary usage and sentence construction patterns between human and AI-generated texts. Our analysis revealed significant differences in vocabulary usage and sentence construction patterns between human and AI-generated texts. The detailed breakdown of these linguistic metrics is presented in Table IV.

b) Key discriminative patterns: Analysis revealed nine primary features distinguishing humans from AI authorship, ranked by discriminative power:

- Lexical Diversity: Human texts exhibit 24% higher type-token ratios and 32% higher MTLD scores, indicating greater vocabulary variation versus AI's repetitive patterns.
- Figurative Language: Human writers use 100% more idioms and 53% more metaphors, demonstrating superior mastery of culturally embedded expressions.
- Error Patterns: Human texts contain significantly more grammatical (300% higher) and spelling errors (375% higher), while AI content shows mechanical accuracy.
- Sentence Variation: Humans demonstrate 34% greater sentence length variability and 23% deeper syntactic structures, creating more natural textual rhythm.
- Discourse Markers: AI uses 31% more explicit connectives ("however," "therefore"), creating artificially structured discourse.

Additional distinguishing features include strategic passive voice usage (humans: 12% vs. AI: 18%), emotional expression patterns (humans show broader sentiment range: $\mu = 0.127 \pm 0.284$ vs. AI: $\mu = 0.089 \pm 0.201$), semantic coherence (humans: 0.68 vs. AI: 0.82, indicating AI's mechanical topic adherence), and conceptual complexity (humans show higher idea complexity: 0.58 vs. AI: 0.52, with greater knowledge domain diversity: 0.63 vs. 0.48).

c) Classification implications: These linguistic differences provide the foundation for our ensemble model's 97.2% accuracy. The systematic patterns - reduced lexical diversity, increased structural rigidity, mechanical error reduction, and formulaic discourse markers in AI content versus human creativity, variability, and natural imperfection - enable reliable automated detection. Cross-validation confirmed that these features maintain discriminative power across different text types and lengths, supporting the practical deployment of real-world content verification applications. Statistical validation using Chi-square tests confirmed all observed differences achieved significance ($p < 0.001$), while effect size analysis revealed considerable practical significance (Cohen's $d > 0.8$) for primary discriminative

features. A comprehensive linguistic analysis establishes the theoretical foundation underlying the effectiveness of our detection methodology.

V. Discussion and Implications

Our experimental results demonstrate significant advances in AI-generated content detection through ensemble learning methodologies. This section synthesizes the key findings, practical implications, and future research directions that emerged from our comprehensive evaluation (Tables V and VI).

1) Key findings and contributions: Our ensemble approach achieved 97.2% accuracy, significantly outperforming individual classifiers and state-of-the-art methods, including transformer-based models (BERT: 94.7%, RoBERTa: 95.3%) and commercial tools (GPTZero: 89.4%, Turnitin: 91.7%). Statistical analysis using McNemar's tests confirmed significance ($p < 0.001$) with large effect sizes (Cohen's $d > 1.3$), establishing genuine methodological advances rather than random variations. The computational efficiency advantage (23ms vs. 187ms for transformer models) demonstrates practical viability for real-world deployment.

The superiority of ensemble modeling validates the integration of multiple classifiers with diverse features compared to individual algorithms. Our linguistic analysis revealed systematic differences between human and AI content: humans exhibit 24% higher lexical diversity, 100% more figurative language usage, and greater syntactic variability, while AI content shows mechanical accuracy, formulaic discourse patterns, and reduced creativity. These findings align with recent research [40] emphasizing the importance of multi-modal approaches in complex text classification tasks.

TABLE V Comprehensive Performance Comparison with State-of-the-Art Methods

Method	Accuracy	Precision	Recall	F1score	Inference
OurEnsemble	97.2%	97.5%	96.4%	97.3%	23
BERT-base(fine-tuned)	94.7%	94.2%	95.1%	94.6%	182
RoBERTa-base(fine-tuned)	95.3%	95.7%	95.2%	95.3%	195
GPT-2 Detector	93.8%	92.9%	91.7%	93.0%	174
GPTZero(Commercial)	89.4%	85.8%	91.2%	88.9%	145
Turnitin AIDetection	91.7%	93.1%	90.2%	91.6%	210
DetectGPT(Academic)	92.1%	92.4%	92.1%	92.1%	167
OpenAI Classifier	90.3%	89.7%	91.0%	90.3%	156

TABLE VI Comprehensive Linguistic Analysis of Human vs. AI Content

Linguistic Dimension	Human Content	AI Content	Difference	Significance
Lexical Features				
Type-Token Ratio	.72	.58	+24%	p <.001
MTLD Index	89.3	67.5	+32%	p <.001
Average Word Length	4.8 ± 2.7	5.2 ± 1.9	-8%	p <.001
Syntactic Features				
Avg. Sentence Length	17.3	19.8	-14%	p <.001
Parse Tree Depth	7.9	6.4	+23%	p <.001
Complex Sentence Ratio	38%	45%	-18%	p < .001
Passive Voice Usage	12%	18%	-50%	p < .001
Discourse Features				
Connective Density	5.2%	6.8%	-31%	p < .001
Lexical Repetition	15.6%	19.3%	-24%	p < .001
Topical Coherence	.68	.82	-21%	p < .001
Stylistic Features				
Idioms per 1000 words	3.8	1.9	+100%	p < .001
Metaphors per 1000 words	5.2	3.4	+53%	p < .001
Cultural References	2.7	1.2	+125%	p < .001
Error Patterns				
Grammar Errors/1000	2.8	.7	+300%	p < .001
Spelling Errors/1000	1.9	.4	+375%	p < .001
Logical Inconsistencies	1.2	.9	+33%	p = .043

2) Practical applications and impact: The framework's balanced performance (97.5% precision, 96.4% recall) makes it suitable for critical applications requiring both accuracy and comprehensive coverage. Educational institutions can leverage our system to maintain academic integrity with minimal false accusations while capturing most AI-generated submissions. News organizations benefit from rapid content authentication (23ms inference time) for verifying article authenticity and preventing misinformation spread. Cybersecurity applications can integrate our models for identifying automated bot-generated content in social engineering attacks, while digital platforms can maintain content authenticity through real-time verification systems.

Cross-domain robustness evaluation confirmed consistent performance across academic papers, social media posts, and technical documentation (95.4% average accuracy), while baseline methods showed significant degradation (transformer models: 88.7%, commercial tools: 82.3%). This generalization capability addresses critical limitations in existing detection systems that struggle with diverse content types.

3) Future research directions: Several promising avenues emerge from our findings: (1) Real-time optimization through model compression and efficient feature extraction for immediate content verification, (2) Multilingual extension adapting our ensemble framework across different linguistic structures and cultural contexts, (3) Hybrid deep learning integration combining transformer architectures with traditional machine learning ensembles for enhanced performance, potentially incorporating insights from graph neural network architectures [41] and ensemble methods proven effective in related domains [42][43], (4) Adversarial robustness investigating model resilience against evasion attacks and developing defense mechanisms, and (5) Ethical frameworks addressing privacy implications, bias mitigation, and transparent disclosure standards for AI-generated content.

The rapid evolution of AI generation capabilities necessitates continuous adaptation of detection methodologies. Our ensemble framework provides a robust foundation for these developments, offering both methodological insights and empirical benchmarks for advancing AI-generated text detection systems. Future work should focus on maintaining detection effectiveness while preserving computational efficiency and ensuring the ethical deployment of these systems across diverse applications.

VI. Conclusion

This work showcases the application of machine learning and natural language processing skills in the human-AI text identification dialectic. Despite the many gaps and shortcomings in the literature that are pointed out in the features of language and models in ensemble learning, the collective classification performed better. Also, the performance of the Gradient Boosting ensemble surpasses that of the Bagging ensemble, which is another argument in favor of advanced machine learning methods for text categorization.

We highlight the features of authorship of the text, which show the authorship of the AI or a person, such as lexical richness, the use of similar phrases, error types and frequency, and variation in the sentence structure. As contributions to the ethics of artificial intelligence and policy, digital forensic science, and moderating systems where the endorsement of content plays a significant role, these facts are noteworthy. This study proposes that future initiatives should expand the refinement of these classification techniques for various types of texts and other fields, thereby addressing the concern of the growing phenomenon of AI text generation.

References

[1] D. Weber-Wulff et al., "Testing of detection tools for AI-generated text," Int. J. Educ. Integr., vol. 19, no. 1, pp. 1–39, 2023.

[2] S. Chakraborty, A. S. Bedi, S. Zhu, B. An, D. Manocha, and F. Huang, "Position: On the Possibilities of AI-Generated Text Detection," Proc. Mach. Learn—Res., vol. 235, pp. 6093–6115, 2024.

[3] V. S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang, and S. Feizi, "Can AI-Generated Text be Reliably Detected?," pp. 1–37, 2023.

[4] Y. Wang, Y. Pan, M. Yan, Z. Su, and T. H. Luan, "A Survey on ChatGPT: AI-Generated Contents, Challenges, and Solutions," IEEE Open J. Comput. Soc., vol. 4, no. July, pp. 280–302, 2023.

[5] E. N. Crothers, N. Japkowicz, and H. L. Viktor, "Machine-Generated Text: A Comprehensive Survey of Threat Models and Detection Methods," IEEE Access, vol. 11, no. May, pp. 70977–71002, 2023.

[6] S. Al Amer, M. Lee, and P. Smith, "Adopting Ensemble Learning for Cross-lingual Classification of Crisis-related Text On Social Media," LoResMT 2024 - 7th Work. Technol. Mach. Transl. Low-Resource Lang. Proc. Work., pp. 50–56, 2024.

[7] Y. Li et al., "MAGE: Machine-generated Text Detection in the Wild," Proc. Annu. Meet. Assoc. Comput. Linguist, vol. 1, pp. 36–53, 2024.

[8] E. Kamateri and M. Salampasis, "An Ensemble Framework for Text Classification," Inf., vol. 16, no. 2, 2025.

[9] S. Sankaranarayanan, A. T. Sivachandran, A. S. Mohd Khairuddin, K. Hasikin, and A. R. Wahab Sait, "An ensemble classification method based on machine learning models for malicious Uniform Resource Locators (URL)," PLoS One, vol. 19, no. 5, pp. 1–20, 2024.

[10] A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," Expert Syst. Appl., vol.. 244, no. May 2023, p. 122778, 2024.

[11] W. Feng, J. Gou, Z. Fan, and X. Chen, "An ensemble machine learning approach for classification tasks using feature generation," Conn: Sci., vol. 35, no. 1, 2023.

[12] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," SN Comput. Sci., vol. 2, no. 3, pp. 1–21, 2021.

[13] K. Krishna, Y. Song, M. Karpinska, J. Wieting, and M. Iyyer, "Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense," Adv. Neural Inf. Process. Syst., vol. 36, no. NeurIPS, pp. 1–32, 2023.

[14] N. B. Reinke, A. L. Parkinson, and G. R. Kafer, "A tutorial activity for students to experience generative artificial intelligence: students' perceptions and actions," Adv. Physiol. Educ., vol. 49, no. 2, pp. 461–470, 2025.

[15] Y. Wang et al., "M4GT-Bench: Evaluation Benchmark for Black-Box Machine-Generated Text Detection," Proc. Annu. Meet. Assoc. Comput. Linguist, vol. 1, pp. 3964–3992, 2024.

[16] G. Huang, Y. Zhang, Z. Li, Y. You, M. Wang, and Z. Yang, "Are AI-Generated Text Detectors Robust to Adversarial Perturbations?" Proc. Annu. Meet. Assoc. Comput. Linguist, vol. 1, pp. 6005–6024, 2024.

[17] K. Kuznetsov et al., "Robust AI-Generated Text Detection by Restricted Embeddings," EMNLP 2024 - 2024 Conf. Empir. Methods Nat. Lang. Process. Find. EMNLP 2024, pp. 17036–17055, 2024.

[18] Y. Cao et al., "A Comprehensive Survey of AI-Generated Content (AIGC): A History of Generative AI from GAN to ChatGPT," J. ACM, vol. 37, no. 4, 2023.

[19] A. Bakhtin, S. Gross, M. Ott, Y. Deng, M. Ranzato, and A. Szlam, "Real or Fake? Learning to Discriminate Machine from Human Generated Text," pp. 1–17, 2019.

[20] F. Khiled and M. S. H. Al-Tamimi, "Hybrid System for Plagiarism Detection on A Scientific Paper," Turkish J. Comput. Math. Educ., vol. 12, no. 13, pp. 5707–5719, 2021.

[21] F. Pedro, M. Subosa, A. Rivas, and P. Valverde, "Artificial Intelligence in Education: Challenges and Opportunities for Sustainable Development Education Sector United Nations Educational, Scientific and Cultural Organization," Minist. Educ., pp. 1–46, 2019.

[22] I. Solaiman et al., "Release Strategies and the Social Impacts of Language Models," 2019.

[23] A. Harada, D. Bollegala, and N. P. Chandrasiri, "Discrimination of human-written and human and machine written sentences using text consistency," Proc. - IEEE 2021 Int. Conf. Comput. Commun. Intell. Syst. ICCCIS 2021, pp. 41–47, 2021.

[24] N. Islam, D. Sutradhar, H. Noor, J. T. Raya, M. T. Maisha, and D. M. Farid, "Distinguishing Human Generated Text From ChatGPT Generated Text Using Machine Learning," 2023.

[25] T. Fagni, F. Falchi, M. Gambini, A. Martella, and M. Tesconi, "TweepFake: About detecting deepfake tweets," PLoS One, vol. 16, no. 5 May, pp. 1–16, 2021.

[26] E. De Santis, A. Martino, F. Ronci, and A. Rizzi, "From Bag-of-Words to Transformers: A Comparative Study for Text Classification in Healthcare Discussions in Social Media," IEEE Trans. Emerg. Top. Comput. Intell., vol. 9, no. 1, pp. 1–15, 2024.

[27] M. Kestemont, K. Luyckx, and W. Daelemans, "Intrinsic plagiarism detection using character trigram distance scores ACCUMULATE-Acquiring Crucial Medical Information using Language Technology View project," no. January 2011.

[28] M. Kuta and J. Kitowski, "Optimisation of Character n-gram Profiles," pp. 500–511, 2014.

[29] A. Mumuni and F. Mumuni, "Automated data processing and feature engineering for deep learning and big data applications: A survey," J. Inf. Intell., vol. 3, no. 2, pp. 113–153, 2024.

[30] T. Masuyama and H. Nakagawa, "Two Step POS Selection for SVM Based Text Categorization," IEICE Trans. Inf. Syst., vol. E87-D, no. 2, pp. 373–379, 2004.

[31] A. Mutsaddi and A. Choudhary, "Enhancing Plagiarism Detection in Marathi with a Weighted Ensemble of TF-IDF and BERT Embeddings for Low-Resource Language Processing," Proc. - Int. Conf. Comput. Linguist. COLING, pp. 89–100, 2025.

[32] H. Allam, L. Makubvure, B. Gyamfi, K. N. Graham, and K. Akinwolere, "Text Classification: How Machine Learning Is Revolutionizing Text Categorization," Inf., vol. 16, no. 2, pp. 0–40, 2025.

[33] A. Gormantara, "Visualization System For Sentiment Analysis Using Textblob On Twitter," Temat. J. Penelit. Tek. Inform. Dan Sist. Inf., pp. 21–26, 2021.

[34] M. P. Behera, A. Sarangi, D. Mishra, and S. K. Sarangi, "A Hybrid Machine Learning algorithm for Heart and Liver Disease Prediction Using Modified Particle Swarm Optimization with Support Vector Machine," Procedia Comput. Sci., vol. 218, no. 2022, pp. 818–827, 2022.

[35] P. P. Putra et al., "Enhancing the Decision Tree Algorithm to Improve Performance Across Various Datasets," vol. 8, no. 2, pp. 200–212, 2024.

[36] O. Kamat, "Plagiarism Detection Using Machine Learning," Int. Res. J. Mod. Eng. Technol. Sci., 2023.

[37] S. Mitrović, D. Andreoletti, and O. Ayoub, "ChatGPT or Human? Detect and Explain. Explaining Decisions of Machine Learning Model for Detecting Short ChatGPT-generated Text," pp. 1–11, 2023.

[38] N. Köbis and L. D. Mossink, "Artificial intelligence versus Maya Angelou: Experimental evidence that people cannot differentiate AI-generated from human-written poetry," Comput. Human Behav., vol. 114, no. September 2020, 2021.

[39] J. Lee, T. Agrawal, A. Uchendu, T. Le, J. Chen, and D. Lee, "PlagBench: Exploring the Duality of Large Language Models in Plagiarism Generation and Detection," 2024.

[40] E. K. Elsayed and D. R. Fathy, "Sign language semantic translation system using ontology and deep learning," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 1, pp. 141–147, 2020.

[41] A. M. Mahmoud, A. S. Desuky, H. Fathy, and H. Abdeldaim, "An Overview and Evaluation on Graph Neural Networks for Node Classification," Int. J. Theor. Appl. Res., vol. 3, no. 1, pp. 379–386, 2024.

[42] E. A. Mahareek, E. K. Elsaid, N. M. El-Desouky, and K. A. El-dahshan, "Survey: Anomaly Detection in Surveillance Videos," Int. J. Theor. Appl. Res., vol. 3, no. 1, pp. 328–342, 2024.

[43] E. A. Mahareek, D. R. Fathy, E. K. Elsayed, N. M. Eldesouky, and K. A. Eldahshan, "Violence Prediction in Surveillance Videos," Appl. Comput. Sci., vol. 20, no. 3, pp. 1–16, 2024.

Reliability Risk Assessment Approaches in Software Engineering: A Review Structured by Software Development LifeCycle (SDLC) Phases and Reliable Sub-Characteristics

Lehka Subramanium, Saadah Hassan, Mohd. Hafeez Osman, Hazura Zulzalil
Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, 43400, Selangor, Malaysia

***Abstract*—Reliability risk is a critical concern in software development, as failures can result in system downtime, degraded performance, data integrity issues, financial losses and loss of user trust. The increasing complexity of modern systems, driven by dynamic workloads, distributed architecture, and unpredictable interactions, amplifies these risks. In regulated industries like healthcare, finance, and transportation, software reliability directly affects safety, compliance and operational continuity, making robust risk assessment essential. Despite recent development and improvement on numerous reliability risk assessment techniques, system failures continue to be potent, creating concerns on scope, applicability and limitations. This paper will dive deep into evaluating recent methods, the advantages and disadvantages of the application itself, while critically assessing the research gaps. Here, the techniques are categorized across the software development lifecycle (SDLC), to bridge methods to phase-specific reliability needs. Consequently, the paper addresses methodological synthesis of recent practices, identifies segments where existing techniques fail to live up to expectations, and summarize future research directions for achieving more robust and adaptive reliability risk assessment.**

***Keywords*—Reliability; risk assessment; SDLC**

I. Introduction

Reliability is a mandatory quality trait in the software industry, ensuring that systems perform persistently and consistently under specific set of conditions without failure [1]. In the modern age where the world has set foot on AI, achieving the utmost level of reliability is increasingly competitive. Though modern software systems are continuously growing, with each development, the complexity of each system is growing. Besides that, integrating distributed architectures, cloud computing, and the Internet of Things (IoT) further prompt dilemma on reliability concerns. Generally, these systems commonly encounter unprecedent workloads, evolving user demands, hardware failures, and security vulnerabilities, all of which create chaos for operation reliability [2]. Key reliability expectations include handling system failures robustly, ensuring fault tolerance, eliminating data corruption, and mitigating downtime [3]. As software systems grow and expand, the complexity of managing reliability risks becomes a more significant burden. Therefore, identifying leading indicators and mitigating them are crucial. Consequently, risk assessment techniques have become a ground-breaking tool for analyzing and mitigating potential reliability issues throughout the software development lifecycle.

A series of risk assessment techniques [4] have been developed to identify and manage reliability risk. In the past. tools such as Failure Mode and Effects Analysis (FMEA), Fault Tree Analysis (FTA), and Reliability Block Diagrams (RBD) have been significant to identify potential points of failure. In recent years, numerous advanced techniques, including probabilistic risk assessments, machine learning models, and simulation-based techniques, have undoubtedly improved the accuracy of predicting, quantifying reliability risks and efficiency of risk management. These approaches have enabled software engineers to detect weaknesses proactively and develop mitigation strategies before risks manifest into system failures.

Despite these advancements, several challenges remain unsolved. No single technique can comprehensively address all dimensions of reliability risk. Many techniques have limited applicability across diverse system architectures, operational environments, or specific failure modes [5]. Some methods are resource-intensive or not scaled for large, distributed systems, while others lack adaptability for the dynamic nature of modern system. Our previous systematic literature review in Subramanium et al. [6], identified key reliability characteristics, associated risks and evaluation metrics, but did not examine how existing risk assessment techniques map to these characteristics across different phases of the software development lifecycle. This gap highlights the need for a structured synthesis of current techniques to identify the coverage limitations and research opportunities.

Unlike prior reviews that often discuss reliability risk assessment techniques in isolation or without considering their relevance across development phases, this study provides a structured synthesis by mapping techniques to both the software development lifecycle (SDLC) phases and reliability sub-characteristics. This dual-perspective approach clarifies where each technique fits best, what risks it addresses, and where methodological gaps persist despite decades of research [7]. Systematically comparing the advantages, limitations, and domain-specific applications of existing methods, this paper goes beyond descriptive reviews to offer a decision-support view for practitioners selecting suitable techniques at different stages of development. For researchers, the findings highlight

underexplored intersections between risk assessment methods, evolving software architectures, and emerging paradigms like cloud computing, IoT, and AI-driven systems [8] providing a foundation for innovative, adaptive reliability assessment frameworks in future work.

The remainder of this paper is organized as follows: Section II provides the background on reliability key concepts. Section III reviews the related work, including existing literature and applications across different system types. Section IV outlines the methodology used to analyze and compare the identified techniques. Section V presents the results, including the classification of techniques, their advantages, limitations and domain specific applications. Next, Section VI discusses the findings, highlighting research gaps and implications for practice and future research. Finally, Section VII concludes the paper by summarizing the key insights and proposing directions for advancing reliability risk assessment practices

II. Background

A. Software Reliability and its Sub-Attributes

A fundamental quality attribute of a product is the reliability of the product itself. Reliability is defined as its capability to perform functions consistently and persistently under specified conditions for a predetermined period [6],[9]. Reliability has been universally recognized in major software quality models such as McCall, Boehm, FURPS, ISO 9126, and the more recent ISO 25010 [10], [11] all of which singled out reliability as a prominent quality attribute and further classify it into several sub-characteristics. In our most recent work, *A Systematic Literature Review on Characteristics Influencing Software Reliability* (2024), we critically summarized a comprehensive analysis of these sub-characteristics and their function on overall software reliability [6]. Here, software reliability is commonly examined through dimensions such as fault tolerance, recoverability, maturity, and availability. Specifically, fault tolerance emphasize the system's ability to continue operating correctly even in the presence of faults [12], whereas recoverability focuses the capacity to restore optimal operations after a failure with minimum damage [13]. Furthermore, maturity affects the system's stability and defect density over time, indicating the effectiveness of its development and testing processes [13],[14],[15]. Next, the availability dimension focuses on making sure the system remains operational and accessible persistently whenever required. This is a crucial factor in safety-critical domains such as finance and healthcare [13]. Digesting these sub-characteristics provides the fundamental basis for evaluating how recent risk assessment techniques address different factors of software reliability across the software development lifecycle.

B. Risk Assessment in Software Engineering

The flow of risk assessment in software engineering commonly involves identifying potential threats, analyzing their probability, prioritizing them, and executing mitigation steps and monitoring strategies [16]. On the scope of software reliability, risks generally sprout out from factors such as complexity in system, unprecedented workloads, distributed architecture, and operational uncertainties that jeopardizes performance and stability [17]. Lately, various techniques have been deployed to analyze and mitigate such risks, ranging from conventional approaches like Failure Mode and Effects Analysis (FMEA), Fault Tree Analysis (FTA), and Reliability Block Diagrams (RBD) to advanced and dynamic methods such as probabilistic risk assessments, Bayesian networks, and simulation-based techniques [18],[19],[20]. Machine learning [21], [22] and AI-driven models [8],[23] have also been explored to improve predictive accuracy and support real-time decision-making in reliability risk assessment. Despite all these efforts, the gaps exist in understanding their applicability, limitations, and compatibility with different stages of software development lifecycle.

C. Need for SDLC-Based Categorization

The Software Development Lifecycle (SDLC) provides a structured and comprehensive framework for planning, designing, developing, testing, and maintaining software systems [7]. Each segment addresses unique reliability risks and acts as a leading indicator to address and tackle risk that may cost catastrophic failures at latter stage of product development. For instance, poor requirements analysis may blind reliability constraints, while deficient testing could allow risk to bypass undetected until late in deployment. Studies indicate that both the choice of development methodology and the factors stressed at each stage significantly amplify reliability outcomes [7]. In particular, [7] highlights fifteen major factors influencing reliability, with decision made in requirement and design during early stages of development having the most significant impact on the final product's quality. This further reiterates the mandatory need to implement reliability risk assessment techniques across all SDLC, instead of applying them only at later stages of development. By narrowing existing techniques to specific SDLC stages, this paper aims to identify coverage gaps and provide perception into how reliability risk assessment practices can be made more structured, comprehensive, effective, context-aware and aligned with phase specific reliability needs.

III. Related Work

The sky is the limit for reliability risk assessment in software engineering. The field continues to evolve over time to cater to increasing complexity of modern systems and the persistent challenge of mitigating software failures. In the most recent times, studies have revealed a wide range of approaches, ranging from foundational qualitative and quantitative methods to advanced machine learning and artificial intelligence-driven techniques. These developments objectives are to enhance the accuracy, comprehensiveness, and real-time applicability of risk assessment across the Software Development Life Cycle (SDLC).

A recent study in 2023 by Jing et al [24] has investigated on the conventional reliability risk assessment techniques, such as Fault Tree Analysis (FTA) and Failure Mode and Effects Analysis (FMEA), which have served as primary tools for risk analysis, especially during the initial design phases. For instance, FTA to employ a top-down deductive approach to identify sequences of events leading to a failure, while FMEA uses a bottom-up inductive method to systematically analyse

potential failure modes and their consequences. Event Tree Analysis (ETA) is often applied alongside these methods to explore the range of possible outcomes following an initiating event. Despite the widespread use for equipment failure diagnosis, root cause analysis, and preliminary safety assessments, these methods face several challenges. A paper in 2022 by Shayan et al [25] evaluated the conventional FMEA, though comprehensive and computationally efficient, struggles with data fragmentation, inconsistent formats, and limitations in the Risk Priority Number (RPN) calculation, which often gives equal weight to severity, occurrence, and detection, sometimes leading to mis prioritized risks. Similarly, FTA encounters difficulties in handling dynamic system characteristics and large-scale data, and all three methods remain largely manual and prone to subjectivity, especially when systems undergo frequent updates.

To further mitigate uncertainty and data incompleteness in traditional methods, researchers have incorporated Fuzzy Logic into risk assessment frameworks, giving rise to Fuzzy Fault Trees (FFT), Fuzzy Event Trees (FET), and Fuzzy FMEA (F-FMEA) [18]. Although conventional techniques focus on precise probabilities, fuzzy methods which enables the use of linguistic variables to shortlist likelihoods and outcomes, making these methods suitable for data-scarce environments. For example, F-FMEA improves conventional RPN calculations by singularly analysing significance, tendency of occurrence, and detection, reducing human impact and enhancing sensitivity to minute parameter alteration. Consequently, integration of fuzzy reasoning has paved ways to a more suitable and adaptive risk prioritization in complex and uncertain environments. A similar approach using the fuzzy FMEA has been applied in the field of cyber security [26] and IT-governance [27] to evaluate risk maturity levels. Building on these successful enhancement, significant research effort continues to improve integration of conventional methods with advanced modelling, automation and analytics techniques for more comprehensive reliability risk assessment [4],[28],[29].

Extending this effort, recent research has focused on addressing randomness and dynamic behavior in systems. For instance, probabilistic techniques such as Markov Chain Models (MCM) and Monte Carlo Simulation (MCS) have been introduced [30],[31],[32],[33]. As for MCM, [30] and [31] have provided a framework for modeling stochastic transitions between system states, while [33] and [34] enables probabilistic exploration of risk scenarios under uncertainty in MCS. However, a specific study on embedded system by [32] discussed that MCS becomes computationally expensive if system complexity and the number of variables increase, limiting its applicability to large-scale systems.

Recently, one other significant advancement of Artificial Intelligence (AI) has opened new opportunities for reliability risk assessment. AI leverage machine learning, deep learning and knowledge representation methods to automate fault detection, predicting failure probabilities and analyse large scale data in real time. With this, new risk assessment techniques were explored by a recent study in 2024 [8], Knowledge Graphs (KGs), in particular, the Fault Knowledge Graphs (FKGs), vector machines and neural network [23]. These AI powered techniques enable adaptive and self-learning reliability risk assessment frameworks. The author has focused on using FKGs to assess key parameters like reliability of the returned information. Similarly, in 2025 [23], Zhao et all has evaluated the use of FKGs to design a multi-level modular structure on an electric power system using neural network and support vector machines by comparing against conventional methods. With all the efforts, a paper in 2023 [35] has critically provided analysis of FKGs that it faces insufficient standardized modelling practices and complexity in extracting information from unstructured data sources. Nevertheless, by further enhancing studies and research on automated techniques and ontology construction, the true potential of KGs in reliability engineering can be realised and it will be groundbreaking.

In a nutshell, the evolution of reliability risk assessment in software engineering shows a shift in pathway from a static framework, toward more dynamic, automated, intelligent, and data-driven techniques. Conventional methods like FTA and FMEA do bring a valuable foundation but tend to get ineffective with dynamic system behaviour and bulk datasets [24]. Increasingly, probabilistic approaches, fuzzy logic, and knowledge graphs address a small sample limitation by introducing probabilistic reasoning, uncertainty modeling, and structured data integration.

Future research should address challenges such as knowledge graph standardization, hybridizing AI and probabilistic methods for improved explainability, and developing computationally efficient techniques for large-scale, real-time applications. Such advancements will be critical for ensuring the reliability and resilience of next-generation software systems

IV. Methodology

This study adopts a structured review approach to analyze software reliability risk assessment techniques across the software development lifecycle (SDLC). The primary objective is to explore how various techniques have been applied, their strengths and limitations, and their suitability for addressing reliability risks in modern software environments.

As illustrated in Fig 1, the review process follows five interconnected stages:

- Step 1: Literature Review and Data Collection
- Step 2: Selection of Relevant Techniques
- Step 3: Mapping Techniques to SDLC Phases
- Step 4: Comparative Analysis of Techniques
- Step 5: Data Synthesis and Research Gaps

Each stage is explained in the following subsections, ensuring a clear and logical flow from gathering information to drawing insights and identifying research gaps.

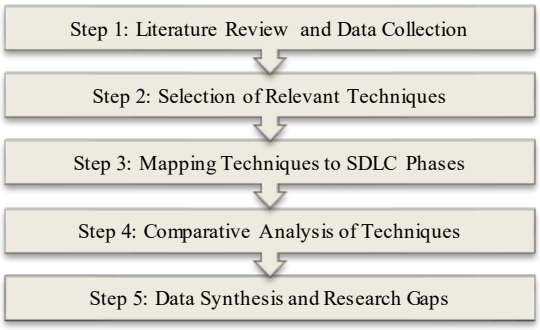


Fig. 1. Design procedure.

A. Literature Review and Data Collection

The first stage involved peer-reviewed studies from academic databases, including IEEE Xplore, Scopus, ACM Digital Library, and Google Scholar. Keywords such as "software reliability," "risk assessment techniques," "reliability sub-characteristics," and "reliability risk analysis" were combined using Boolean operators to ensure comprehensive coverage. Also, terms such as "reliability AND risk matrix", "reliability AND Probabilistic Risk Assessment" and with other risk assessment techniques uniquely were employed to retrieve studies emphasizing reliability-related risk modelling methods. This approach minimized irrelevant results and ensured that only techniques applicable to software reliability evaluation were captured.

To maintain relevance, only studies published in English between 2015 and 2025 were included. Non-peer-reviewed sources, duplicate studies, and research lacking methodological detail were excluded.

B. Selection of Relevant Techniques

The next step was to select the techniques that were most frequently discussed and demonstrated clear relevance to software reliability risk assessment. A simple rating approach was used to give a more objective selection basis.

Each technique was rated on aspects such as:

- Frequency of application in studies
- Coverage of reliability sub-characteristics (referring to the standard quality model, ISO 25010)
- Clarity and completeness of methodological description
- Reported advantages and limitations

This allowed the review to prioritize techniques with strong academic support while still discussing fewer common methods for completeness.

C. Mapping Techniques to SDLC Phases

Furthermore, techniques were classified according to the phases of the Software Development Life Cycle (SDLC) to provide practitioners with a clear reference on which techniques are most suitable for specific stages of software development, bridging the gap between research insights and real-world application.

D. Comparative Analysis of Techniques

Once the techniques were aligned to SDLC phases, a comparative analysis was undertaken to present the information in a structured and transparent manner. Specifically, each technique was analyzed across the following dimensions:

- Risk assessment techniques: Name of the technique
- Overview: Brief description of its approach or methodology
- Domain application: Software domains or system types where it has been applied (example: IoT, cloud computing, real-time systems, AI)
- Coverage of reliability sub-characteristics: Mapping to reliability characteristics based on software quality model ISO 25010.
- Advantages: Reported strengths such as accuracy, scalability, cost-effectiveness.
- Limitations- Key challenges like implementation complexity or lack of empirical validation.
- References: Key studies supporting the analysis

This classification ensures a comprehensive, phase-oriented perspective for evaluating the suitability of risk assessment techniques in modern software environments, enabling both researchers and practitioners to select methods best aligned with their system requirements.

E. Data Synthesis and Research Gaps

After the comparative analysis, findings were synthesized to identify patterns, trends, and gaps. This stage helped highlight gaps in the literature, such as the lack of empirical validation for certain techniques or limited application in specific software domains, providing directions for future studies.

Despite its structured approach, this review is limited by potential publication bias due to reliance on peer-reviewed studies in English and selected academic databases. Furthermore, the analysis depends on secondary data without empirical testing, which may restrict the generalizability of conclusions. Acknowledging these limitations provides transparency and highlights opportunities for empirical validation in future studies.

V. Results

Table I presents a phase-wise summary of software reliability risk assessment techniques across the analysis, development, and testing stages of the software development, and testing stages of the software development lifecycle (SDLC). Categorizing techniques by SDLC phase ensures that the distinct risk management requirements of each stage are adequately addressed, enabling practitioners to apply the most suitable methods at the right time. Building on this table, the subsequent discussion evaluates each technique in terms of overview, domain application, coverage of reliability sub characteristics, providing a comprehensive understanding of their strengths, weaknesses for modern software environments.

TABLE I SUMMARY OF SOFTWARE RELIABILITY RISK ASSESSMENT TECHNIQUES ACROSS SDLC PHASES

Risk Assessment Techniques	Overview	Domain Application	Coverage of reliability sub-attributes	Advantages	Limitations	References
REQUIREMENTS						
Risk Matrix	A qualitative or semi-quantitative tool that categorizes risks by severity and likelihood to prioritize them.	- Safety-Critical Systems - Industrial Automation & Manufacturing - Cloud Computing - Real-time and Embedded systems - Software Engineering	Availability, Maturity, Recoverability	- A transparent representation of risk levels using likelihood and consequence scales - Low implementation cost - Support quick decision making - Customizable (project size, different domains, risk categories) - Communication friendly	- Less efficient in modeling complex reliability behaviors. - Subjectivity in risk scoring - Lack of probabilistic measures - Limited predictive capability for dynamic or real-time environments where risk levels evolve quickly	[36], [37], [38]
Risk-Driven Requirement Engineering	A systematic approach to derive and prioritize requirements based on identified risks.	- Safety-Critical (aerospace, healthcare) - Large-scale enterprise software telecommunication system) -Agile and Incremental Development	Maturity, Fault Tolerance:	- Early risk identification - Improve requirement quality - Support requirement prioritization - Reduce rework costs.	- Dependent on expert judgement - Time-consuming for large projects - Limited tool integration - Subjective risk estimation	[20], [39]
Failure-Oriented Requirements Engineering (FORE)	Focus on identifying failure modes and their impact on requirements to mitigate risks early.	- Safety-critical - Large scale industrial systems - Real-time systems - IoT applications	Maturity Fault Tolerance	- Failure identification at the requirement phase - Support early fault analysis - Integrates with safety standards - Support systematic traceability from failures to requirements	- Require detailed system knowledge early in the system development - Time intensive for complex systems with many failures modes - Subjectivity on expert driven failure modelling	[40]
Goal-Oriented Risk Assessment (GORA)	Identifies risks through the lens of system goals, ensuring alignment between objectives and risk mitigation.	- Complex socio-technical system - Business critical software - Safety systems - Requirement engineering - Decision support systems - Policy driven IT systems	Maturity	- Improve traceability from risks to system objectives - Useful in multi-stakeholder environments - Support what-if analysis - Provides structured modeling linking risks to goals, obstacles and mitigations.	- Limited reliability quantitative evaluation - High effort for large systems with many goals - Require tools for efficient goal modeling.	[41], [42]

Risk-Based Requirement Prioritization	Prioritize requirements based on their associated risks to ensure critical risks are addressed first.	- Agile projects, incremental development, time constrained software releases -Cloud software, e-commerce platforms, enterprise applications	Maturity Fault Tolerance	- Ensure critical, high-risk requirements are developed first - Cost-effective - Fits well with Agile and iterative methodologies like Scrum - Support risk-driven release planning	- May ignore low-risk requirements that could be important for reliability - Subjective prioritization - Limited support for quantitative risk modelling	[43], [33]
Probabilistic Risk Assessment (PRA) in Requirements	Quantifies the likelihood of risks in meeting requirements by analyzing statistical probabilities	- Nuclear system - Defense system - Autonomous vehicles - Mission critical software - Model-based systems engineering for safety and reliability certifications	Maturity Availability Fault Tolerance	- Provide quantitative, probabilistic measures of risk. - Support early decision making under uncertainty using likelihood and consequences modelling - integrates with Fault Tree Analysis (FTA) and Event Tree Analysis (ETA) - Enables reliability-based requirement verification early	- Require extensive data for accurate probability estimation - High modelling complexity and computational costs for large systems - Steep learning curve	[44], [33], [45], [46]
DESIGN						
Hazard and Operability Study (HAZOP)	A structured and systematic technique to identify hazards and operability issues in processes or systems by evaluating deviations from design intent.	- Process control system - Chemical plants - Industrial automation - Safety critical software	Fault Tolerance, Availability	- Structured approach ensures no risk scenario is overlooked. - Suitable for early phases, enabling proactive risk mitigation. - Adopted in safety standards	- Require time-consuming workshops for comprehensive coverage. - Largely qualitative - Subjective to expert knowledge accuracy.	[47], [48],
Fault Tree Analysis (FTA)	A top-down, deductive approach to identify causes of system-level failures by analyzing fault logic.	- Aerospace, nuclear power plants - Automotive safety systems.	Availability, Maturity	- Easy-to-understand fault visualization. - Supports probabilistic failure estimation using failure rate data. - Standardized technique.	- Assumes independence of failures, unrealistic for complex systems. - Static analysis, cannot handle event sequences well. - Large systems produce complex, unwieldy trees.	[18], [35], [29]
Event Tree Analysis (ETA)	A forward-looking technique that models the possible outcomes of an initiating event through branching event sequences.	- Nuclear safety - Aviation risk modeling - Chemical plants.	Availability:	- Forward looking approach complements FTA - Support probability mapping	- Event paths multiply quickly in complex systems - Quality depends on correct events - Assumption on event independence	[18], [24]
Reliability Block Diagram (RBD)	Graphical representation of system reliability structure by connecting components in series or	- Aerospace - Manufacturing - Power Systems	Availability, Fault Tolerance	- Provide graphical modeling	- Assumes static configuration	[15]

	parallel to depict dependency			- Suitable to identify critical components	- Limited for dynamic or stochastic systems	
Monte Carlo Simulation	Uses random sampling and statistical modeling to simulate the probability of different outcomes and assess risk.	- Finance - Cloud Systems - Telecommunication	Fault Tolerance, Availability, Maturity	- Captures uncertainty and probabilistic behavior - Handles complex, interdependent systems	- Requires high computational cost - Accuracy depends on number of simulations	[33], [34]
Markov Analysis and Modelling	Analyzes system states and transitions using probabilistic modeling for systems with random state changes over time.	- Embedded Systems, - Network Systems, - Health Systems	Availability, Recoverability, Fault Tolerance	- Models stochastic and time-dependent failures - Supports repairable system modeling	- Assumes exponential failure distributions	[44], [46] [30], [32], [31],
Petri Nets	A graphical and mathematical modeling tool to represent concurrent, distributed, and dynamic system processes.	- Industrial Control Systems - Distributed Systems	Fault Tolerance, Recoverability	- Models stochastic and time-dependent failures -Supports repairable system modeling	- Modeling complexity increases with system size	[4], [49]
Stochastic Petri Nets (SPN)	Extends Petri Nets with stochastic timing to model system behaviors and performance under uncertainty.	- Real-time Systems - IoT, - Cyber-Physical Systems	Fault Tolerance, Availability, Recoverability	- Captures stochastic behavior and timing aspects - Supports dynamic analysis	- Higher computational cost; requires specialized expertise	[50]
Bayesian Networks	A probabilistic graphical model that represents variables and their dependencies using directed acyclic graphs.	- Autonomous Vehicles - Smart Grids, Healthcare	Fault Tolerance, Availability, Recoverability, Maturity	- Probabilistic reasoning under uncertainty -Supports learning from data and evidence updating	- Scalability issues for large systems. - Data dependency specification challenges	[4],[51]
Fuzzy Logic	Mathematical approach for handling uncertainty and imprecision in risk assessment.	- IoT - Edge Computing - Decision-Support Systems	Fault Tolerance, Recoverability, Maturity	- Handles imprecise, linguistic, or uncertain data well - Suitable for expert-judgment-based analysis	- Subjectivity in membership function design - Lacks standardization across domains -	[18]
DEVELOPMENT						
Failure Modes and Effects Analysis (FMEA)	A systematic approach to identify potential failure modes, their causes, and effects to prioritize corrective actions.	Fault Tolerance, Availability, Recoverability	- Automotive - Aerospace - Industrial Systems	- Structured and systematic - Early defect identification	- Time-consuming - qualitative in nature - Subjective to expert knowledge	[24], [29]
Code Review and Static Code Analysis	Evaluates code quality by manually reviewing or using automated tools to detect bugs, vulnerabilities, and design flaws.	Maturity, Fault Tolerance	- Software Development - Web Applications like Banking Systems	- Early defect detection before runtime - Automated tools available - Cost-effective for initial phases	- Limited to static properties - Cannot detect runtime or dynamic behavior issues	[52]
Reliability Growth Models	Models that predict system reliability improvement over time as defects are identified and corrected during testing.	Maturity, Availability	- Telecommunication Systems	- Focus reliability improvement over testing cycles - Quantitative assessment of defect detection	- Assumes failure patterns follow statistical distributions	[53], [54], [55]
Dynamic Analysis and	Examine system behavior during execution to identify	Fault Tolerance, Availability, Recoverability	- Cloud Systems - IoT	- Provides real-time insights	- Performance overhead	[56]

Runtime Monitoring	potential runtime errors, resource bottlenecks, or reliability risks.		- Autonomous Systems	- Detects runtime anomalies and adaptive risks	- May be complex to implement at scale	
Test Case Prioritization and Reliability Testing	Prioritizes test cases based on their likelihood of revealing faults and evaluates reliability through systematic testing.	Fault Tolerance, Availability, Maturity	- Embedded Systems, Telecom	- Ensures critical functionalities tested first - Improves fault detection efficiency	- Effectiveness depends on quality of test cases - May require domain-specific customization	[43], [57]
Model-Based Reliability Assessment	Uses formal models to simulate and evaluate system reliability based on its design, configuration, and usage scenarios.	Fault Tolerance, Availability, Recoverability, Maturity	- Safety-Critical Systems (Aerospace, Healthcare)	- Formal and structured modeling - Supports early-stage risk evaluation - Allows simulation of different scenarios	- High modeling effort - Requires specialized skills and accurate system specifications	[33], [50], [58]
TESTING						
Reliability Growth Testing using the Software Reliability Growth Model (SRGM) technique	Evaluates the improvement in system reliability by statistical models that estimate the reliability of software systems by analyzing failure data over time to predict future reliability.	Maturity, Fault Tolerance	- Software Development - Telecommunications - Defense	- Quantitative evaluation of reliability improvement over time	- Requires historical failure data - Ineffective to dynamic changing environments	[53], [59]
Fault Injection Testing	Deliberately introduces faults into a system to evaluate their fault tolerance and recovery mechanisms.	Fault Tolerance, Recoverability	- Cloud Systems - Automotive - Safety-Critical Software	- Evaluates system robustness under failure conditions - Reveals hidden vulnerabilities	- May be costly or risky in production-like environments	[60]
Model-Based Reliability Testing (MBT)	Uses formal models of system behavior to derive test cases and evaluate reliability.	Fault Tolerance, Availability, Maturity	- Aerospace, - Healthcare - Embedded Systems	- Systematic test generation from formal models - Improves coverage and defect detection	- High modeling effort - Requires formal specifications	[61], [62]
Reliability prediction models	Uses mathematical models to estimate system reliability based on design parameters and historical data.	Fault Tolerance, Availability, Maturity	- Defense - Aerospace - Software Engineering	- Supports early estimation of reliability before deployment - Quantitative risk assessment	- Accuracy depends heavily on data quality and assumptions	[22]
Scenario-based testing for reliability	Focuses on testing specific scenarios to evaluate reliability under predefined conditions.	Availability, Fault Tolerance, Recoverability	- Automotive - IoT - Real-Time Systems	- Captures real-world operational conditions - Effective for stress and edge-case testing	- Scenario design complexity - May not cover unexpected runtime behaviors	[63]
Fault Knowledge Graphs	Knowledge-driven approach where information about system faults, failure modes, causes, and their interrelationships is represented as a graph structure	Availability, Fault Tolerance, Recoverability	- AI Systems - Cloud Computing - Complex Software Architectures	- Enables fault pattern analysis using structured knowledge representation - Supports root cause analysis	- Still emerging - Limited standardization; requires high-quality knowledge bases	[35]

VI. Discussion

The categorization of reliability risk assessment techniques across software development lifecycle (SDLC) phases, as summarized in Table I, provides a structured perspective on when and how these methods are applied by linking techniques to specific phases, analysis, design, development, and testing. It also enables practitioners to select methods that align with the unique risk management needs of each stage, avoiding the common pitfall of relying on a one-size-fits-all approach.

In the requirement phase, techniques such as Risk Matrix, Risk-Driven Requirement Engineering, Failure-Oriented Requirements Engineering (FORE), Goal-Oriented Risk Assessment (GORA), Risk-Based Requirement Prioritization, and Probabilistic Risk Assessment (PRA) enable early-stage identification and prioritization of risks.

Next, in the design phase, techniques such as Hazard and Operability Study (HAZOP), Fault Tree Analysis (FTA) , Event Tree Analysis (ETA), Reliability Block Diagrams (RBD) [15], Monte Carlo Simulation, Markov Models, Petri Nets [35], Stochastic Petri Nets (SPN), Bayesian Networks [64], and Fuzzy Logic provide deeper insights into system behavior and potential failure paths.

Further, the development phase introduces Failure Mode and Effects Analysis (FMEA), Code Review and Static Code Analysis, Reliability Growth Models [54], Dynamic Analysis and Runtime Monitoring, Test Case Prioritization and Reliability Testing, and Model-Based Assessment to ensure software reliability as code implementation progresses [4].

Lastly, in the testing phase, as systems transition into validation, techniques like Software Reliability Growth Models (SRGM) [59], Fault Injection, Model-Based Reliability Testing, Reliability Prediction Models, Scenario-Based Testing for Reliability, and Fault Knowledge Gaps Analysis ensure comprehensive risk evaluation under realistic operational conditions.

Despite the availability of diverse risk assessment techniques across the SDLC, one major research gap [65] lies in the fragmented treatment of reliability sub-characteristics [64]. Many techniques either focus on overall system reliability without distinguishing between sub-characteristics such as fault tolerance, availability, recoverability, and maturity, or they emphasize only one aspect while neglecting others [4]. For example, Fault Tree Analysis (FTA) and Reliability Block Diagrams (RBD) primarily evaluate system failure probabilities and availability but provide little insight into recoverability or fault tolerance mechanisms. Conversely, Bayesian Networks and Stochastic Petri Nets (SPN) are strong in modeling fault tolerance and dynamic behavior but are rarely extended to evaluate maturity or maintainability, leading to incomplete risk profiles. This fragmented coverage hinders practitioners from understanding how different reliability sub-attributes interact, for instance, a system might have high fault tolerance yet fail to recover quickly from unexpected outages, compromising operational continuity [66].

Another critical gap concerns the domain-specific nature of reliability requirements. Many existing techniques were designed for general-purpose reliability analysis and do not adequately address domain-driven reliability priorities [67]. For example, recoverability is crucial in financial systems where transaction consistency and disaster recovery are essential, whereas fault tolerance and real-time failure detection dominate autonomous vehicle and industrial automation domains. Similarly, availability and service continuity are top priorities in telecommunication networks and cloud platforms, where downtime directly impacts service quality and revenue. However, most current methods fail to provide customized reliability assessment frameworks that align with the unique risk profiles, operational constraints, and regulatory requirements of these domains [68].

Moreover, as emerging domains such as AI-driven applications, IoT ecosystems, and cyber-physical systems introduce dynamic operational behaviors, data uncertainty, and real-time decision-making, conventional methods like Reliability Growth Models struggle to capture the evolving risk landscape. This highlights [12] the need for adaptive, domain-aware risk assessment approaches that integrate multiple sub-characteristics, model interdependencies, and align reliability priorities with the specific operational and safety requirements of diverse application areas.

VII. Conclusion and Future Work

The study highlighted the evolution of reliability risk assessment techniques across software lifecycle phases, revealing significant progress in modeling, analysis, and prediction. However, findings indicate that despite methodological diversity, many approaches remain phase-specific, static, or computationally demanding, limiting their adaptability to dynamic, data-rich software environments. Moreover, the absence of standardized evaluation benchmarks, limited domain-specific tailoring, and poor alignment with modern development practices such as agile and DevOps hinder practical adoption.

Future research should focus on developing integrated, adaptive, and benchmarked reliability frameworks that balance accuracy, scalability, and usability while accommodating the complexity and dynamism of modern software systems. Emphasis on real-time analytics, AI-driven risk prediction, automated scenario simulation, and cross-domain customization will be essential. Furthermore, extending research toward edge computing, cyber-physical systems, and autonomous platforms offers opportunities to validate and refine techniques under safety-critical and resource-constrained conditions, ensuring reliability assurance remains robust in next-generation software landscapes.

Acknowledgment

We would like to thank Universiti Putra Malaysia for all the support given.

References

[1] L. C. Hao, L. J. Wu, R. Yan, X. Y. Han, and L. L. Tang, “Research on Software Reliability Index Allocation Method Based on Network Architecture,” Proc. 2019 Int. Conf. Qual. Reliab. Risk, Maintenance, Saf. Eng. QR2MSE 2019, no. Qr2mse, pp. 551–556, 2019, doi: 10.1109/QR2MSE46217.2019.9021200.

[2] Y. Zhao, P. Li, J. Deng, M. Gao, Z. Wang, and X. Fan, “Reliability Evaluation Method of Distribution Network Considering Differential Reliability Requirements of End Users,” Proc. 2021 IEEE 4th Int. Electr. Energy Conf. CIEEC 2021, pp. 3–8, 2021, doi: 10.1109/CIEEC50170.2021.9510553.

[3] P. Rotella and S. Chulani, “SRC ratio method: Benchmarking software reliability,” Proc. - 2017 IEEE 28th Int. Symp. Softw. Reliab. Eng. Work. ISSREW 2017, pp. 61–64, 2017, doi: 10.1109/ISSREW.2017.75.

[4] S. Kabir and Y. Papadopoulos, “Applications of Bayesian networks and Petri nets in safety, reliability, and risk assessments: A review,” Saf. Sci., vol. 115, no. November 2018, pp. 154–175, 2019, doi: 10.1016/j.ssci.2019.02.009.

[5] J. Ai, W. Su, and F. Wang, "Software Reliability Evaluation Method Based on a Software Network," Proc. - 29th IEEE Int. Symp. Softw. Reliab. Eng. Work. ISSREW 2018, pp. 136–137, 2018, doi: 10.1109/ISSREW.2018.00-15.

[6] L. Subramanium, S. Hassan, M. H. Osman, and H. Zulzalil, "A Systematic Literature Review on Characteristics Influencing Software Reliability," Int. J. Informatics Vis., vol. 8, no. 4, pp. 2344–2353, 2024, doi: 10.62527/joiv.8.4.3665.

[7] V. Yakovyna, M. Seniv, and I. Symets, "The Relation between Software Development Methodologies and Factors Affecting Software Reliability," Int. Sci. Tech. Conf. Comput. Sci. Inf. Technol., vol. 1, pp. 377–381, 2020, doi: 10.1109/CSIT49958.2020.9321937.

[8] G. Spasova and D. Dinev, "Exploring the Effectiveness and Reliability of Artificial Intelligence," CIEES 2024 - IEEE Int. Conf. Commun. Information, Electron. Energy Syst., pp. 1–5, 2024, doi: 10.1109/CIEES62939.2024.10811341.

[9] S. Yin, Q. Shi, Y. Wang, and C. Chen, "Summary of software reliability Research," IOP Conf. Ser. Mater. Sci. Eng., vol. 1043, no. 5, 2021, doi: 10.1088/1757-899X/1043/5/052039.

[10] T. Hovorushchenko, "The software emergent properties and them reflection in the non-functional requirements and quality models," Proc. Int. Conf. Comput. Sci. Inf. Technol. CSIT 2015, no. September, pp. 146–153, 2015, doi: 10.1109/STC-CSIT.2015.7325454.

[11] P. Nistala, K. V. Nori, and R. Reddy, "Software quality models: A systematic mapping study," Proc. - 2019 IEEE/ACM Int. Conf. Softw. Syst. Process. ICSSP 2019, pp. 125–134, 2019, doi: 10.1109/ICSSP.2019.00025.

[12] K. E. De Souza and F. C. Ferrari, "A Systematic Review of Fault Tolerance Techniques for Adaptive and Context-Aware Systems," Proc. - 2022 IEEE Int. Conf. Auton. Comput. Self-Organizing Syst. ACSOS 2022, pp. 21–30, 2022, doi: 10.1109/ACSOS55765.2022.00020.

[13] E. Ismail, N. Utelieva, A. Balmaganbetova, and S. Tursynbayeva, "The choice of measures reliability of the software for space applications," 2nd Int. Conf. Electr. Commun. Comput. Eng. ICECCE 2020, no. June, pp. 12–13, 2020, doi: 10.1109/ICECCE49384.2020.9179411.

[14] X. H. Wang, Y. X. Li, W. H. Fan, J. Q. Xuan, L. Z. Wang, and M. M. Mu, "Evaluation of product maturity based on quality characteristic," IEEE Int. Conf. Ind. Eng. Eng. Manag., vol. 2016-January, pp. 742–746, 2016, doi: 10.1109/IEEM.2015.7385746.

[15] J. Lucas, A. Thiraviam, A. Elshennawy, and A. M. Albar, "The effectiveness of reliability programs and tools based on design maturity and complexity," Proc. - Annu. Reliab. Maintainab. Symp., pp. 1–5, 2017, doi: 10.1109/RAM.2017.7889658.

[16] S. D. S. Lopes, I. G. Vargas, A. L. De Oliveira, and R. T. V. Braga, "Risk Management for System of Systems: A Systematic Mapping Study," Proc. - 2020 IEEE Int. Conf. Softw. Archit. Companion, ICSA-C 2020, pp. 258–265, 2020, doi: 10.1109/ICSA-C50368.2020.00050.

[17] J. Zhang, "Field Product Reliability Risk Assessment," Proc. - Annu. Reliab. Maintainab. Symp., vol. 2022-Janua, pp. 1–6, 2022, doi: 10.1109/RAMS51457.2022.9894003.

[18] C. Quan, L. Lingqiang, Y. Dongping, and Y. Gu, "A reliability risk analysis method based on the fuzzy fault tree and fuzzy event tree," Proc. 2016 11th Int. Conf. Reliab. Maintainab. Saf. Integr. Big Data, Improv. Reliab. Serv. Pers. ICRMS 2016, 2017, doi: 10.1109/ICRMS.2016.8050082.

[19] X. Pan and M. Zhang, "Quality and Reliability Improvement Based on the Quality Function Deployment Method," Proc. - 12th Int. Conf. Reliab. Maint. Safety, ICRMS 2018, pp. 38–42, 2018, doi: 10.1109/ICRMS.2018.00018.

[20] O. T. Arogundade, S. Misra, O. O. Abayomi-Alli, and L. Fernandez-Sanz, "Enhancing Misuse Cases with Risk Assessment for Safety Requirements," IEEE Access, vol. 8, pp. 12001–12014, 2020, doi: 10.1109/ACCESS.2019.2963673.

[21] M. Banga, A. Bansal, and A. Singh, "Implementation of Machine Learning Techniques in Software Reliability: A framework," 2019 Int. Conf. Autom. Comput. Technol. Manag. ICACTM 2019, pp. 241–245, 2019, doi: 10.1109/ICACTM.2019.8776830.

[22] C. Ji, "Reliability Evaluation and Prediction of Mechanical System Based on Machine Learning Technology," IEEE 1st Int. Conf. Ambient Intell. Knowl. Informatics Ind. Electron. AIKIIE 2023, pp. 1–5, 2023, doi: 10.1109/AIKIIE60097.2023.10390487.

[23] Z. Zhang and X. Wang, "Design and Research of an Electric Power System Reliability Evaluation Model Based on Artificial Intelligence Algorithms," 2025 5th Asia-Pacific Conf. Commun. Technol. Comput. Sci., pp. 149–155, 2025, doi: 10.1109/ACCTCS66275.2025.00034.

[24] X. Hu, J. Liu, H. Dou, H. Chen, and Y. Zhang, "Automatic Generation of Component Fault Trees from AADL Models for Design Failure Modes and Effects Analysis," IEEE Int. Conf. Softw. Qual. Reliab. Secur. QRS, pp. 550–561, 2023, doi: 10.1109/QRS60937.2023.00060.

[25] S. Kumar Akula, H. Salehfar, and S. Behzadirafi, "Comparision of Traditional and Fuzzy Failure Mode and Effects Analysis for Smart Grid Electrical Distribution Systems," 2022 North Am. Power Symp. NAPS 2022, pp. 1–6, 2022, doi: 10.1109/NAPS56150.2022.10012165.

[26] N. A. Chandra, A. A. Putri Ratna, and K. Ramli, "Development of a cyber-situational awareness model of risk maturity using fuzzy fmea," 2020 Int. Work. Big Data Inf. Secur. IWBIS 2020, pp. 127–136, 2020, doi: 10.1109/IWBIS50925.2020.9255543.

[27] U. Yudatama and R. Sarno, "Evaluation maturity index and risk management for it governance using Fuzzy AHP and Fuzzy TOPSIS (case Study Bank XYZ)," 2015 Int. Semin. Intell. Technol. Its Appl. ISITIA 2015 - Proceeding, pp. 323–327, 2015, doi: 10.1109/ISITIA.2015.7220000.

[28] J. Cooper, "Implementation of the Product Reliability Program," 2024 Pan Pacific Strateg. Electron. Symp. Pan Pacific 2024, pp. 1–6, 2024, doi: 10.23919/PanPacific60013.2024.10436446.

[29] P. Fithri, N. A. Riva, L. Susanti, and B. Yuliandra, "Safety analysis at weaving department of PT. X Bogor using Failure Mode and Effect Analysis (FMEA) and Fault Tree Analysis (FTA)," 2018 5th Int. Conf. Ind. Eng. Appl. ICIEA 2018, pp. 382–385, 2018, doi: 10.1109/IEA.2018.8387129.

[30] Y. Jun, J. Chenyu, X. Zhihui, L. Mengkun, and Y. Ming, "Markov/CCMT_ Towards an integrated platform for dynamic reliability and risk analysis," Process Saf. Environ. Prot., vol. 155, pp. 498–517, 2021, doi: 10.1016/j.psep.2021.09.043.

[31] M. F. Aly, I. H. Afefy, R. K. Abdel-Magied, and E. K. A. Elhalim, "A comprehensive model of reliability, availability, and maintainability (RAM) for industrial systems evaluations," Jordan J. Mech. Ind. Eng., vol. 12, no. 1, pp. 59–67, 2018.

[32] S. Shu, Y. Wang, and Y. Wang, "An approach to architecture-based fault tolerance evaluation with fault propagation," Proc. 2015 1st Int. Conf. Reliab. Syst. Eng. ICRSE 2015, vol. 56, pp. 3–9, 2015, doi: 10.1109/ICRSE.2015.7366478.

[33] Y. Jie, W. Wang, G. Wang, and M. Zhang, "Reliability assessment of high microgravity science experiment system based on probabilistic risk assessment," Proc. 2016 Progn. Syst. Heal. Manag. Conf. PHM-Chengdu 2016, pp. 1–6, 2017, doi: 10.1109/PHM.2016.7819797.

[34] M. I. Lunesu, R. Tonelli, L. Marchesi, and M. Marchesi, "Assessing the risk of software development in agile methodologies using simulation," IEEE Access, vol. 9, pp. 134240–134258, 2021, doi: 10.1109/ACCESS.2021.3115941.

[35] L. Shen, H. Tang, L. Wang, J. Cai, and X. Cui, "A Fault Knowledge Graph Creation Method and Application based on Fault Tree Analysis and Failure Mode, Effects and Criticality Analysis," Proc. 2023 IEEE 3rd Int. Conf. Inf. Technol. Big Data Artif. Intell. ICIBA 2023, vol. 3, no. Iciba, pp. 35–39, 2023, doi: 10.1109/ICIBA56860.2023.10165591.

[36] M. Blumenschein, J. Spasic, J. Steckert, and J. Uythoven, "An Approach to Reliability Assessment of Complex Systems at CERN," Proc. - Annu. Reliab. Maintainab. Symp., vol. 2019-Janua, pp. 1–6, 2019, doi: 10.1109/RAMS.2019.8769004.

[37] P. Souvannalath, S. Premrudeepreechacharn, and K. Ngamsanroaj, "Determining Power Transformer Maintenance Plan Using Three-Dimensional Risk Matrix: Note: Sub-titles are not captured in Xplore and should not be used," 2022 IEEE Int. Conf. Power Syst. Technol. Embrac. Adv. Technol. Power Energy Syst. Sustain. Dev. POWERCON 2022, pp. 1–6, 2022, doi: 10.1109/POWERCON53406.2022.9929632.

[38] Z. Danlin, J. Han, S. Jialiang, and Y. Lin, "A risk assessment approach based on fuzzy 3D risk matrix for network device," 2016 2nd IEEE Int.

Conf. Comput. Commun. ICCC 2016 - Proc., pp. 1106–1110, 2017, doi: 10.1109/CompComm.2016.7924876.

[39] F. Wiesweg, A. Vogelsang, and D. Mendez, "Data-driven Risk Management for Requirements Engineering: An Automated Approach based on Bayesian Networks," Proc. IEEE Int. Conf. Requir. Eng., vol. 2020-Augus, pp. 125–135, 2020, doi: 10.1109/RE48521.2020.00024.

[40] P. Garraghan et al., "Emergent Failures: Rethinking Cloud Reliability at Scale," IEEE Cloud Comput., vol. 5, no. 5, pp. 12–21, 2018, doi: 10.1109/MCC.2018.053711662.

[41] F. Başak Aydemir, P. Giorgini, and J. Mylopoulos, "Multi-objective risk analysis with goal models," Proc. - Int. Conf. Res. Challenges Inf. Sci., vol. 2016-Augus, 2016, doi: 10.1109/RCIS.2016.7549302.

[42] D. Alrajeh, A. Van Lamsweerde, J. Kramer, A. Russo, and S. Uchitel, "Risk-driven revision of requirements models," Proc. - Int. Conf. Softw. Eng., vol. 14-22-May-, pp. 855–865, 2016, doi: 10.1145/2884781.2884838.

[43] C. Hettiarachchi and H. Do, "A Systematic Requirements and Risks-Based Test Case Prioritization Using a Fuzzy Expert System," Proc. - 19th IEEE Int. Conf. Softw. Qual. Reliab. Secur. QRS 2019, pp. 374–385, 2019, doi: 10.1109/QRS.2019.00054.

[44] R. G. Maidana, T. Parhizkar, A. Gomola, I. B. Utne, and A. Mosleh, "Supervised dynamic probabilistic risk assessment : Review and comparison of methods," Reliab. Eng. Syst. Saf., vol. 230, no. May 2022, p. 108889, 2023, doi: 10.1016/j.ress.2022.108889.

[45] F. M. Safie, R. G. Stutts, and Z. Huang, "Reliability and probabilistic risk assessment - How they play together," in Proceedings - Annual Reliability and Maintainability Symposium, 2015, vol. 2015-May, pp. 1–5, doi: 10.1109/RAMS.2015.7105058.

[46] Q. Liu, L. Xing, and C. Wang, "Framework of Probabilistic Risk Assessment for Security and Reliability," Proc. - 2017 IEEE 2nd Int. Conf. Data Sci. Cyberspace, DSC 2017, pp. 619–624, 2017, doi: 10.1109/DSC.2017.35.

[47] F. U. Muram, M. A. Javed, and S. Punnekkat, "System of Systems Hazard Analysis Using HAZOP and FTA for Advanced Quarry Production," 2019 4th Int. Conf. Syst. Reliab. Safety, ICSRS 2019, pp. 394–401, 2019, doi: 10.1109/ICSRS48664.2019.8987613.

[48] R. A. Viegas, F. de A. da S. Mota, A. P. C. S. Costa, and F. F. P. dos Santos, "A multi-criteria-based hazard and operability analysis for process safety," Process Saf. Environ. Prot., vol. 144, pp. 310–321, 2020, doi: 10.1016/j.psep.2020.07.034.

[49] J. Hu and Y. Cao, "Fuzzy Petri net based dynamic risk analysis of complex system considering protection layers," 2015 12th Int. Conf. Fuzzy Syst. Knowl. Discov. FSKD 2015, pp. 308–312, 2016, doi: 10.1109/FSKD.2015.7381959.

[50] P. Grimmeisen, A. Morozov, T. Fabarisov, A. Wortmann, and C. H. Koo, "Automated Model-Based Reliability Assessment of Software-Defined Manufacturing," IEEE Int. Conf. Emerg. Technol. Fact. Autom. ETFA, vol. 2022-Septe, pp. 1–4, 2022, doi: 10.1109/ETFA52439.2022.9921704.

[51] D. Liu, X. Xu, K. Ma, L. Tao, and M. Suo, "Fault Diagnosis Based on Fault Tree and Bayesian Network with Grey Optimization," Proc. 34th Chinese Control Decis. Conf. CCDC 2022, pp. 1787–1792, 2022, doi: 10.1109/CCDC55256.2022.10033578.

[52] A. M. Stanciu and H. Ciocârlie, "Analyzing Code Security: Approaches and Tools for Effective Review and Analysis," Int. Conf. Electr. Comput. Energy Technol. ICECET 2023, no. November, pp. 1–6, 2023, doi: 10.1109/ICECET58911.2023.10389326.

[53] Q. Li and C. Mao, "Considering Testing-Coverage and Fault Removal Efficiency Subject to the Random Field Environments with Imperfect Debugging in Software Reliability Assessment," Proc. - 2016 IEEE 27th Int. Symp. Softw. Reliab. Eng. Work. ISSREW 2016, pp. 257–263, 2016, doi: 10.1109/ISSREW.2016.13.

[54] C. Jackson, "Reliability growth and demonstration: The Multi-Phase Reliability Growth Model (MPRGM)," Proc. - Annu. Reliab. Maintainab. Symp., vol. 2016-April, pp. 1–6, 2016, doi: 10.1109/RAMS.2016.7447983.

[55] Z. S. Li and D. Xu, "A bayesian approach for modeling reliability growth," Proc. - Annu. Reliab. Maintainab. Symp., vol. 2020-Janua, 2020, doi: 10.1109/RAMS48030.2020.9153616.

[56] T. Sutter, T. Kehrer, M. Rennhard, B. Tellenbach, and J. Klein, "Dynamic Security Analysis on Android: A Systematic Literature Review," IEEE Access, vol. 12, no. April, pp. 57261–57287, 2024, doi: 10.1109/ACCESS.2024.3390612.

[57] P. Vats, A. Gossain, and M. Mandot, "SARLA - A 3-Tier Architectural Framework Based on the ACO for the Probablistic Analysis of the Regression Test Case Selection and their Prioritization," ICRITO 2020 - IEEE 8th Int. Conf. Reliab. Infocom Technol. Optim. (Trends Futur. Dir., pp. 681–687, 2020, doi: 10.1109/ICRITO48877.2020.9198020.

[58] D. Ling, B. Liu, and S. Wang, "A component-based software reliability assessment method considering component effective behavior," 2017 2nd Int. Conf. Reliab. Syst. Eng. ICRSE 2017, no. Icrse, 2017, doi: 10.1109/ICRSE.2017.8030748.

[59] K. Okumoto, A. Asthana, and R. Mijumbi, "BRACE: Cloud-based software reliability assurance," Proc. - 2017 IEEE 28th Int. Symp. Softw. Reliab. Eng. Work. ISSREW 2017, pp. 57–60, 2017, doi: 10.1109/ISSREW.2017.48.

[60] Y. Xu and S. He, "Avionics Equipment Cable Fault Injection Design and Simulation Testing Validation," 2023 Glob. Reliab. Progn. Heal. Manag. Conf. PHM-Hangzhou 2023, pp. 1–5, 2023, doi: 10.1109/PHM-HANGZHOU58797.2023.10482733.

[61] R. Ramler and C. Klammer, "Enhancing Acceptance Test-Driven Development with Model-Based Test Generation," Proc. - Companion 19th IEEE Int. Conf. Softw. Qual. Reliab. Secur. QRS-C 2019, pp. 503–504, 2019, doi: 10.1109/QRS-C.2019.00096.

[62] C. B. Nielsen, P. G. Larsen, J. Fitzgerald, J. Woodcock, and J. Peleska, "Systems of systems engineering: Basic concepts, model-based techniques, and research directions," ACM Comput. Surv., vol. 48, no. 2, 2015, doi: 10.1145/2794381.

[63] M. Jahan, Z. S. H. Abad, and B. Far, "Detecting Emergent Behavior in Scenario-Based Specifications using a Probabilistic Model," Proc. - 10th Int. Model. Requir. Eng. Work. MoDRE 2020, pp. 31–38, 2020, doi: 10.1109/MoDRE51215.2020.00010.

[64] Y. Liu, M. Lu, and B. Xu, "Software reliability case development method based on software reliability characteristic model and measures of defect control," Proc. IEEE Int. Conf. Softw. Eng. Serv. Sci. ICSESS, vol. 0, pp. 1–6, 2016, doi: 10.1109/ICSESS.2016.7883004.

[65] L. Fraccascia, I. Giannoccaro, and V. Albino, "Resilience of complex systems: State of the art and directions for future research," Complexity, vol. 2018, 2018, doi: 10.1155/2018/3421529.

[66] J. Zhang, Y. He, Y. Xie, and A. Zhang, "Product Key Reliability Characteristics Identification Approach Using GA in Manufacturing Process," 2021 Glob. Reliab. Progn. Heal. Manag. PHM-Nanjing 2021, 2021, doi: 10.1109/PHM-Nanjing52125.2021.9612881.

[67] B. Liao, Y. Ali, S. Nazir, L. He, and H. U. Khan, "Security Analysis of IoT Devices by Using Mobile Computing: A Systematic Literature Review," IEEE Access, vol. 8, pp. 120331–120350, 2020, doi: 10.1109/ACCESS.2020.3006358.

[68] T. S. Sakriwala, V. Pandey, and R. K. S. Raveendran, "Reliability Assessment Framework for Additive Manufactured Products," 2020 Int. Conf. Comput. Perform. Eval. ComPE 2020, pp. 350–354, 2020, doi: 10.1109/ComPE49325.2020.9200078.

WiTS: A Wi-Fi-Based Human Action Recognition via Spatio-Temporal Hybrid Neural Network

Pengcheng Gao*
School of Cyber Security, Gansu University of Political Science and Law, Lanzhou 730070, Gansu, China

***Abstract*—Human action recognition has many applications in different scenarios. With the advancement of wireless sensing and the widespread deployment of Wi-Fi devices, the perception technology of Wi-Fi channel state information (CSI) has shown great potential. Related studies identified actions by capturing specific attenuation and distortion features caused by human posture on CSI. These methods are less susceptible to the effects of lighting and object occlusion. However, they have yet to adequately extract information within CSI. The challenge of enhancing model performance through the comprehensive utilization of information features within different dimensions remains an imperative area. To address this, a spatio-temporal hybrid neural network model named WiTS is proposed. It integrates the advantages of different neural networks, using CNN to extract spatial features, combining TCN and Bi-LSTM for dual temporal dimension modeling, and incorporating Transformer's global attention mechanism to achieve comprehensive extraction and multi-level fusion of spatio-temporal features. Additionally, this study further optimizes the original WiTS model from three aspects. The Experiment on WiAR and CSIAR datasets show that the model achieves average accuracy rates of 95.75% and 96.71%, respectively, with F1-scores exceeding 96%. The model has only 2.19 million parameters and less than 560 million FLOPs, offering significant advantages in terms of lightweight design, making it suitable for deployment on limited-computing edge terminals while meeting real-time requirements.**

Keywords*—Wi-Fi CSI; human action recognition; deep learning

I. Introduction

The advent of wireless sensing technology has led to significant advancements in perception technology based on Wi-Fi CSI, which has demonstrated considerable potential. It can be applied to many fields, with the most notable being human action recognition [1-2]. For instance, in the field of smart homes, wireless sensing could recognize users' hand gestures to control electronic devices. In the gaming field, action capture can be used to manipulate characters in real time to enhance the gaming experience. In the field of security, CSI changes in monitored areas can be captured to detect anomalies. In the medical and elderly care fields, wireless sensing technology can be applied for respiratory monitoring, fall detection, etc. [3]. A Wi-Fi-based human action recognition method has been developed that can detect human movements in a signal coverage environment. It utilizes sensors to collect data such as CSI and received signal strength (RSS). This method is distinct from both visual perception methods and those that employ wearable devices for perception, and it possesses three primary advantages. Firstly, the Wi-Fi-based perception is unaffected by lighting conditions and can be used in any lighting environment. Secondly, Wi-Fi signals are less susceptible to obstruction by objects and can penetrate obstacles such as walls, furniture, and appliances, enabling accurate recognition of human movements behind obstructions. Thirdly, unlike visual data collection methods, this approach abstracts human movements by identifying the specific disturbances they cause to signals, offering inherent advantages in terms of privacy protection and making it suitable for deployment in private spaces such as bedrooms and bathrooms [4]. At present, methods for human action recognition using Wi-Fi sensing could be broadly classified into two categories: machine learning methods and deep learning methods. The content below will introduce pertinent research on action recognition based on these two methods.

In the field of machine learning-based methods, numerous researchers have explored this domain. As posited by He et al. [5], the WiG system was the inaugural system to utilize CSI for this task. The Birge-Massart filter was utilized to denoise the signal, thereby preserving significant information. The Local Outlier Factor and Support Vector Machine (SVM) were employed to extract feature data and perform action recognition. In a related study, Zhang et al. [6] extracted domain-independent gesture features at lower signal levels. They proposed a general method applicable to different environments that effectively achieves cross-domain recognition. Xian et al. [7] modified the K-means algorithm to facilitate the recognition of fine-grained actions. Dang et al. [8] combined the Dynamic Time Warping (DTW) algorithm and SVM to match and recognize different actions, effectively improving recognition accuracy. The amplitude and phase difference of subcarrier levels in wireless signals was found to be correlated with human actions by Hao et al. [9]. They combined K-means and Bagging algorithms to optimize SVM, achieving the recognition task at a low computational cost. Huang et al. [10] discovered that Nonlinear Phase Error Variation (NLPEV) data in CSI exhibits good stability and sensitivity to actions, and utilized it to achieve effective HAR in Co-channel Interference (CCI) scenarios. Chelli et al. [11] developed a machine learning framework for action recognition based on average Doppler shift, using KNN for action classification, achieving good action recognition performance. Cheng et al. [12] utilized CSI phase differences to construct an extended matrix for action feature extraction and employed a Gaussian mixture-hidden Markov model to identify CSI feature data. This approach has been shown to

reduce system computation time and improve the fault tolerance rate of segmentation.

In contrast to machine learning, which necessitates the laborious process of manual feature extraction, methods based on deep learning have undergone rapid development in recent years. Zou et al. [13] employed CNNs to extract the most significant features from signals for the purpose of action recognition. Muaaz et al. [14] proposed a system named Wi-Sense, which employs CSI ratio methods to mitigate noise and phase shift effects. This is followed by the generation of spectral maps from preprocessed data and the training of a CNN model using spectral images. Huan et al. [15] proposed a novel activity segmentation method utilizing signal variance differences between action and non-action segments to achieve a balance between robustness and property. Duan et al. [16] proposed a sorting algorithm based on subcarrier correlation and inversion to extract different user signals. The authors combined a bidirectional GRU with an attention mechanism and a convolutional neural network to achieve multi-user action recognition. Meng et al. [17] employed a sparse recovery approach for identifying the primary paths affected by human activities, and constructed a matrix based on the phase differences between adjacent antennas. Thereafter, they proposed a bidirectional GRU structure with an attention mechanism for automatic learning and feature extraction from the matrix. Gao et al. [18] developed a geometric representation of different human hand gestures and used a backtracking search algorithm to recognize them. Cui et al. [19] employed CNN to extract features between different subcarriers and proposed an integrated architecture comprising multiple layers of perception, random forests, and SVM to improve recognition accuracy.

In summary, there are several challenges associated with using machine learning methods for action recognition. Manual feature extraction requires significant labor costs and is subject to the loss of implicit yet crucial information, which complicates the effective differentiation of similar actions. Furthermore, the efficacy of this method is contingent upon the availability of high-quality data; in the absence of such data, the recognition accuracy cannot attain a high level. While deep learning methods have demonstrated efficacy in the extraction of CSI, existing approaches generally utilize spatial features in signals, failing to effectively extract features from the temporal domain. Therefore, developing a model which could parse the rich radio signal strength and phase variation information contained in CSI signals to extract features from both the temporal and spatial dimensions remains an urgent problem to be solved. To address this issue, this study proposes a Wi-Fi-based human action recognition model using a spatio-temporal hybrid neural network. The primary contributions of this paper are as follows:

1) A hybrid architecture model, designated as WiTS, was developed, incorporating TCN, CNN, Bi-LSTM, and Transformer networks, which can effectively extract and fuse the temporal and spatial feature information in Wi-Fi CSI to achieve high-precision human action recognition.

2) The structure of the CNN module was optimized by using a spatial attention block to enhance important spatial locations and replacing traditional convolutions with partial convolutions [20] to reduce parameters.

3) An attention mechanism was employed into the Bi-LSTM to enhance the sensitivity and expressive capability to key information segments.

4) The Sparrow Search Algorithm (SSA) [21] was employed for hyperparameter optimization, further enhancing the performance of model.

The rest part is divided into five sections: Section II introduces related research. Section III provides a comprehensive illustration of the improved methodology. Section IV chiefly presents the experimental configuration and dataset. Section V shows the results of ablation experiments and comparative experiments. Section VI summarizes the entire work.

II. Related Works

To address the issues previously mentioned regarding Wi-Fi CSI in human action recognition and achieve more comprehensive extraction of features, this paper utilizes four deep learning models, CNN, TCN, Bi-LSTM, and Transformer, to construct a hybrid architecture. The subsequent content will offer a concise overview of these seminal networks.

Convolutional Neural Network (CNN) is a type of deep learning model that has been specifically developed for processing grid-like information, which includes convolution, pooling, and fully connected layers. Convolution layers obtain features through local receptive fields and weight sharing. Pooling layers like max pooling and average pooling reduce data dimensions and enhance translation invariance. Fully connected layers are employed for final classification or regression tasks [22]. The advantage of CNNs lies in their hierarchical feature learning capability, where shallow convolutions capture low-level features, while deep networks extract high-level semantic features [23]. In recent years, CNNs have achieved significant progress in image recognition, object detection and other domains, thus becoming an integral component of deep learning research.

Temporal Convolutional Network (TCN) is a network to process time series data. The main structure of TCN principally comprises causal convolution, dilated convolution, and residual connection structure. It ensures temporal dependency through causal convolution; combines dilated convolution to expand the receptive field, thereby effectively capturing long-term dependencies; and introduces residual connections to moderate the training difficulties of deep networks [24]. Compared to traditional recurrent neural networks (RNNs), TCN has faster training speeds and more stable gradients, and they perform exceptionally well in various temporal tasks [25].

Long Short-Term Memory (LSTM) is a type of RNN to address the long-term dependency issue inherent in general RNNs [26]. LSTM introduces three gating mechanisms and the concept of a cell state, in which information can be added, deleted, or modified. The forget gate decides which information will be removed from the cell state. The input

gate decides which new information will be added to it. The function of the output gate is to decide which information will be output from it. During the update process, the cell state is primarily updated through element-wise multiplication and addition, and gradients are linearly propagated along the time axis during backpropagation. This enables LSTM to reliably learn long-term dependencies in sequences and effectively mitigate the gradient vanishing or exploding issues that plague traditional RNNs. Bidirectional Long Short-Term Memory (Bi-LSTM) was proposed by Mike Schuster and Kuldip K. Paliwal [27]. This network consists of a forward LSTM and a backward LSTM connected in parallel. The forward layer reads the sequence in time order to obtain the preceding information, while the backward layer reads the sequence in reverse time order to obtain the subsequent information. This allows the network to use both past and future context, furthering the accuracy of tasks such as classification and prediction.

The Transformer model was first proposed by Vaswani et al. [28] in 2017 and completely varied the paradigm of natural language processing. In contrast to conventional RNN and CNN, the Transformer utilizes attention mechanisms as its primary approach to capture global dependencies in input sequences. This enables parallel computing and facilitates more efficient modeling of long-range dependencies. The core structure includes multi-head attention and position-wise feed-forward networks, and it also incorporates residual connections and layer normalization to optimize the training process. In contemporary applications, the Transformer has been employed not only in natural language processing but also in a variety of downstream tasks, including computer vision, speech processing, and multimodal learning through hybrid architectures that integrate the Transformer with other models [29].

III. Improved Methodology

This paper develops a hybrid architecture network named WiTS based on CNN, TCN, Bi-LSTM, and Transformer to fuse temporal and spatial feature information. The network structure is shown in Fig. 1.

For spatial feature extraction, a CNN branch is utilized to construct a feature encoder that captures spatial local correlations of the input data through local connections and weight sharing. For temporal feature extraction, the temporal encoder is constructed using dilated causal convolutions from TCN to leverage its exponentially increasing receptive field and capture temporal sequence features. Simultaneously, a Bi-LSTM branch utilizes its gated memory mechanism to explicitly model short-term to long-term dependencies in the temporal dimension. This complements TCN's feature extraction and forms a dual-temporal dimension modeling approach that combines parallel capture with recursive memory. To fully examine the deep correlations between different features, the spatio-temporal features extracted by the three branches are integrated and import to the Transformer that uses its self-attention mechanism to establish global dependencies. This architecture complements the spatial feature extraction capabilities of CNN, the temporal feature extraction capabilities of TCN and Bi-LSTM, and the global context modeling capabilities of Transformer, thereby achieving a more comprehensive and deeper feature representation of complex spatio-temporal patterns.

This paper also introduces three improvements based on this model, namely improvements to the LSTM module, improvements to the CNN module, and hyperparameter optimization. The subsequent subsections will provide a detailed introduction to the optimization content.

A. LSTM Module Improvement

In comparison with the conventional Bi-LSTM module, this model incorporates an attention mechanism, assigning distinct weights to the output of each time step of the LSTM, enabling the model to concentrate on the most critical segments in the sequence rather than simply relying on the last hidden state. Fig. 2 shows the optimized LSTM module. The calculation process of the attention mechanism is as follows.

Assume that the output of Bi-LSTM is:

$$X = [h_1, h_2, \cdots, h_T] \tag{1}$$

Where $h_t \in R^{256}$, $X \in R^{B\times T\times 256}$, and 256 is the feature dimension of each time step output of Bi-LSTM. Subsequently, the output h_t of each time step undergoes linear transformation and Tanh activation:

$$e_t = \tanh(w^T h_t + b) \tag{2}$$

Where $w \in R^{256}$ and $b \in R$.

After the transformation and activation of the output of each time step, the attention scores of all time steps are to be concatenated:

$$e = [e_1, e_2, \cdots, e_T] \in R^{B\times T} \tag{3}$$

For each sample, all time step scores are normalized using Softmax to obtain attention weights:

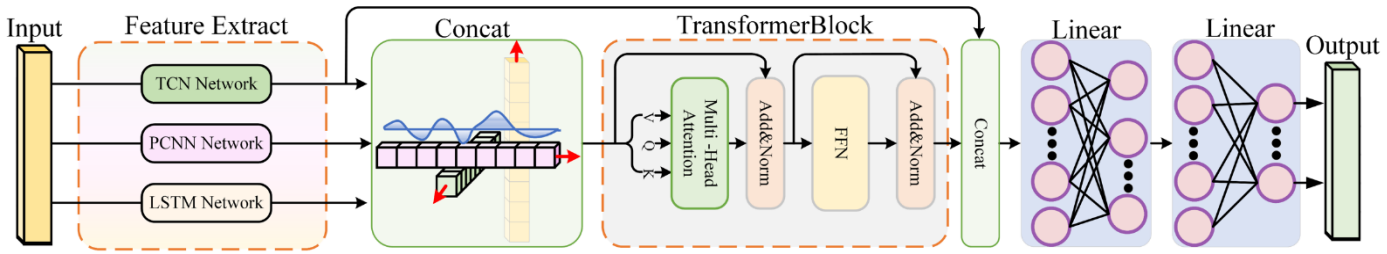


Fig. 1. The structure of WiTS.

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T}\exp(e_k)} \quad (4)$$

where, $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_T] \in R^{B\times T}$.

Weight the total outputs of all time steps of the Bi-LSTM using attention weights to get the global feature vector of the whole sequence after attention weighting:

$$s = \sum_{t=1}^{T} \alpha_t h_t \quad (5)$$

where, $s \in R^{B\times 256}$, h_t is the output of Bi-LSTM at time step t, and α_t is the attention weight at that time step.

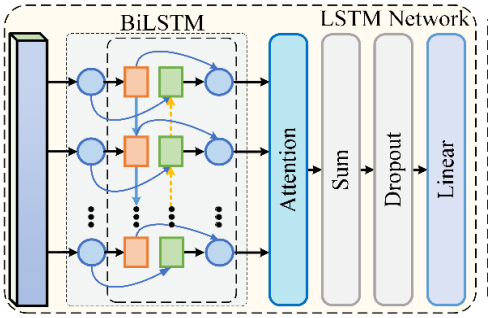


Fig. 2. The structure of optimized Bi-LSTM module.

The calculation process reveals that, in contrast to the conventional Bi-LSTM module, which utilizes the final moment as the output, the enhanced Bi-LSTM prioritizes time steps that are designated as crucial by the attention mechanism within the sequence. It also integrates information from all-time steps within the sequence, thereby offering a more comprehensive approach. This design effectively improves the ability to model long sequences and complex dependencies, and it also increases the richness and discriminative power of feature expression.

B. CNN Module Improvement

To reduce the inference time, partial convolution is utilized in CNN instead of the traditional convolution structure in CNN networks. Partial convolution was proposed by Jierun Chen et al [20]. They believe that although the information in the feature maps to be detected is gradually extracted and aggregated by the model as the depth of the convolutional neural network increases, the feature maps of different layers often contain a lot of redundant information. Consequently, they enhanced the convolution block to reduce the processing of such repetitive information. The structure of Partial Convolution is shown in Fig. 3. The primary implementation method involves the combination of a $k \times k$ convolution with c_p channels and a 1×1 convolution with $c - c_p$ channels to form a hammer-shaped convolution structure, which replaces the traditional $k \times k$ convolution structure with c channels. In this convolution form, only a portion of the channels utilize $k \times k$ convolution, while the remaining portions are processed using 1×1 convolution kernels. This approach has been demonstrated to markedly reduce parameters necessary for convolution operations, thereby enabling more efficient feature extraction. Additionally, it allows for the adjustment of the proportion of two types of convolutions to minimize potential loss of useful feature information.

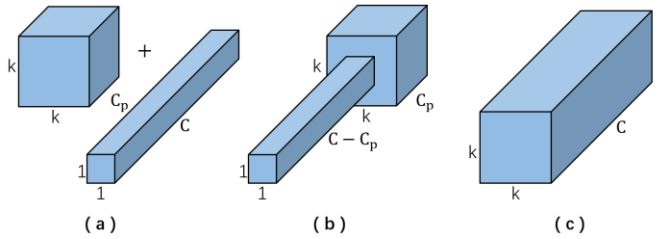


Fig. 3. (a) Convolutional variants; (b) A convolutional framework consisted of one PConv, and one 1*1 Conv; (c) One regular convolutional structure.

To enhance the weights of important spatial locations in the input feature map while suppressing unimportant regions, a spatial attention mechanism module was added to the CNN module, whose structure is shown on the left part of Fig. 4. The primary working principle is as follows: First, the average and maximum values are calculated for the input feature map along the channel dimension, yielding two single-channel feature maps. Second, these two maps are concatenated to form a two-channel feature map. Third, a convolutional layer is utilized to reduce the two-channel feature map to a one-channel feature map, thereby obtaining a spatial attention weight map. The weights are then normalized to the range of 0 to 1 using a Sigmoid activation function. This weight map is then multiplied by the original feature map to enhance important spatial locations.

This improved CNN module is called the PCNN module, and its structure is shown on the right part of Fig. 4.

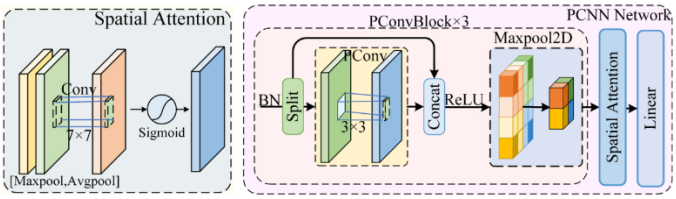


Fig. 4. The structure of PCNN module.

C. Hyperparameter Optimization

The SSA was utilized for the purpose of hyperparameter optimization, with the objective of further enhancing the model's detection capabilities. SSA is a swarm intelligence optimization method inspired by sparrows' foraging and anti-predation behaviors. The algorithm simulates the collaboration and division of labor among sparrows during the process of searching for food and evading predators to solve complex optimization problems. The algorithm first randomly generates a set of sparrow individuals, with each individual representing a solution. The classification of sparrows is typically divided into three distinct categories: The roles of producer, scrounger, and scout are delineated. Producers are responsible for global search and guiding the population's movement. Scroungers follow Producers and perform local search. Scouts are tasked with escaping local optima. After each iteration, the fitness of each sparrow (i.e. the quality of the solution) is evaluated, and roles and positions are adjusted accordingly. This process is repeated iteratively until the maximum number of iterations is reached or convergence conditions are met. The derivation process of the sparrow search algorithm is illustrated below.

Initialization of the population is the first step. Assuming a population size of N, the position of each sparrow is represented as $X_i = (X_{i1}, X_{i2}, \cdots X_{id})$, where d is the dimension of the problem. The initialization formula is as follows:

$$X_i^0 = LB + rand() \times (UB - LB) \quad (6)$$

In this formula, LB and UB represent the lower and upper bounds of the variable, respectively, and $rand()$ denotes a random number within the interval [0,1].

The number of producers is calculated as $PD \times N$, where PD is typically set within the range of 0.2 to 0.3. The position update formula is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot T}\right), R_2 < ST \\ X_{i,j}^t + Q \cdot L, \quad R_2 \geq ST \end{cases} \quad (7)$$

In this formula, $X_{i,j}^t$ denotes the position of the ith sparrow in the jth dimension at generation t. The constant α is typically equal to 1, T signifies the maximum iterations, R_2 is a number in the range [0, 1], ST is the safety threshold, which is typically equal to 0.8, Q is a number following a standard distribution, and L is a vector of all ones. In scenarios where $R_2 < ST$, the sparrow is in a safe state and engages in a global search employing exponential decay. In scenarios where, $R_2 \geq ST$, the sparrow perceives a threat and employs Gaussian perturbation to augment its capacity to evade local optima.

The number of scroungers is $(1 - PD) \times N$, and the position update is based on Formula 8.

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst,j}^t - X_{i,j}^t}{i^2}\right), & f_i > f_g \\ X_{i,j}^t + \left|X_{i,j}^t - X_{best,j}^t\right| \cdot \frac{A^+}{A^T \cdot A}, & f_i = f_g \end{cases} \quad (8)$$

$X_{worst,j}^t$ represents the position of the worst sparrow in the jth dimension in the tth generation, $X_{best,j}^t$ represents the position of the best sparrow in the jth dimension in the tth generation, f_i is the fitness of the ith sparrow, and f_g is the global optimal fitness. Q is a random number following a normal distribution, A is a random vector following a normal distribution, and A^+ is the Moore-Penrose generalized inverse of A. When the follower's fitness is worse than the global optimum, it moves away from the worst individual to avoid getting stuck in a local optimum. When the fitness equals the global optimum, it moves closer to the best individual to enhance development capabilities.

The scouts randomly select a portion of the scroungers, with the number being $SD \times N$. The value of SD is typically between 0.1 and 0.2. The following formula is for position updating:

$$X_{i,j}^{t+1} = X_{best,j}^t + \beta \cdot \left|X_{i,j}^t - X_{best,j}^t\right| \quad (9)$$

Among these, $X_{i,j}^{t+1}$ represents the position of the ith sentinel in the jth dimension in the $t+1$ generation, $X_{best,j}^t$ represents the position of the globally optimal individual in the jth dimension in the tth generation, $X_{i,j}^t$ represents the position of the ith sentinel in the jth dimension in the tth generation, and β is a random number following a normal distribution. When the population becomes stuck in a local optimum, the scouts randomly jump to new positions near the optimal individual, thereby increasing the chance of discovering new optimal solutions and enhancing the diversity of the global search. After each iteration, the fitness of each sparrow is calculated, and the roles of producers, scroungers, and scouts are dynamically adjusted based on fitness rankings to enhance the algorithm's global optimization capability and convergence speed. The application of SSA to the task of optimizing deep learning model hyperparameters involves the mapping of these parameters to the SSA search space and the definition of a fitness function, thereby enabling an automatic search for the optimal combination.

Subsequent experiment results demonstrate that these three optimizations based on the original model have all had positive effects, and that this hybrid architecture significantly improves performance in human action recognition tasks compared to a single network structure.

IV. Sensing Principle and Environment Configuration

A. Wi-Fi Sensing Principle

In an indoor environment, a Wi-Fi sensing system is composed of two primary components: a Wi-Fi signal transmitter (Tx) and a Wi-Fi signal receiver (Rx). When Wi-Fi signals propagate through stationary objects within a room, such as floors, ceilings, and furniture, the angles and paths of reflection remain relatively stable. Consequently, in the absence of external interference, the CSI signals received at each time stamp remain largely unchanged. However, in environments characterized by dynamic targets, such as individuals walking or performing specific actions indoors, the position and posture of the human body undergo fluctuations at distinct temporal intervals. This causes channel interference such as amplitude attenuation and phase distortion when the Wi-Fi signal propagates to the human body, resulting in changes to the CSI received at the receiver. Channel interference caused by the same action is generally similar, while channel interference caused by different actions varies significantly [30].

Wi-Fi technology adheres to the IEEE 802.11 standard, employing OFDM to modulate signals. OFDM divides the spectrum into multiple mutually orthogonal subcarriers, which do not interfere with each other. This frequency division method enables the modulated signal to resist multipath propagation and interference, thereby improving signal robustness. Therefore, this method is widely used in the field of wireless communication. The used dataset is comprised of data collected from wireless network cards that are following this standard. Through Wi-Fi CSI signals, detailed amplitude and phase data can be obtained. This data can be used for channel state analysis to understand the multipath effects and interference conditions during signal propagation.

The process of Wi-Fi signal transmission is shown in Formula 10:

$$\vec{Y} = H\vec{X} + n \quad (10)$$

$\vec{Y}$ means the received signal vector, $\vec{X}$ means the transmitted signal vector, H denotes CSI, which describes the channel frequency response of the subcarriers between the transmitter and receiver, and n denotes the noise vector.

The CSI can be expressed by the following formula:

$$H = \begin{bmatrix} H_{1,1} & \cdots & H_{1,M} \\ \vdots & \ddots & \vdots \\ H_{N,1} & \cdots & H_{N,M} \end{bmatrix} \quad (11)$$

M represents the number of subcarriers. Typically, a larger bandwidth corresponds to a greater number of subcarriers. $H_{N,M}$ is the CSI signal transmitted by the Mth subcarrier at time N.

The CSI measurement value for the Mth subcarrier at time t can be expressed by Formula 12:

$$H_t = |H_t| e^{j\angle H_t} \quad (12)$$

$|H_t|$ represents the amplitude of the Mth subcarrier at that moment, and $\angle H_t$ represents its phase data.

CSI can be further decomposed into static and dynamic components. The former is caused by stationary objects in the environment, while the latter is caused by human movements. Therefore, CSI can also be expressed in the formula below:

$$H = \sum_{s=1}^{S} \alpha_s(f_k) e^{-j2\pi f_k \tau_s} + \sum_{d=1}^{D} \alpha_d(f_k, t) e^{-j2\pi f_k \tau_d(t)} \quad (13)$$

The first part represents the static component, where S denotes the number of static paths, $\alpha_s(f_k)$ denotes the static attenuation coefficient, and τ_s denotes the time delay of the static path. The second part represents the dynamic component, where D denotes the number of dynamic paths, $\alpha_d(f_k, t)$ denotes the dynamic attenuation coefficient, and $\tau_d(t)$ denotes the time delay of the dynamic path. Unlike the static component, the dynamic component varies with time, meaning that changes in the dynamic component occur during the period from the start of an action to its completion. Therefore, Wi-Fi-based action recognition primarily achieves action classification by identifying the characteristic changes in the dynamic components of different actions.

B. Experimental Environment and Dataset

All experiments in this research were conducted under identical conditions to eliminate the effect of environmental factors on the results. The operating environment and main parameter configurations are shown in Table I and Table II.

In this study, two public datasets, CSIAR [30] and WiAR [31], were selected to train the model and verify its recognition performance. The categories of human actions included in the datasets are shown in Table III.

The WiAR dataset was collected in a meeting room measuring 6m × 8m, with 4m between the transmitter and receiver, and a distance of 1m between the volunteer performing the actions and the transmitter. The data collection was carried out by three volunteers, each performing 16 types of coarse-grained actions, with each action repeated 30 times, resulting in a total of 1,440 samples. This study selected 10 of these actions for experimentation, including 7 upper-limb actions, 2 lower-limb actions, and 1 full-body action. The CSIAR dataset was collected in an office environment, with 3m between the transmitter and receiver. Six volunteers participated in the experiment, with each volunteer performing one activity within 20 seconds, repeated 20 times. The 6 actions in this dataset are all coarse-grained actions and are all full-body actions. Overall, the WiAR dataset primarily focuses on upper-body actions and lacks full-body actions, while the 6 actions covered by CSIAR are all full-body actions, effectively supplementing the missing types in the former and better reflecting the model's ability to recognize various human movements.

Since each action in the dataset is executed multiple times, the CSI signals obtained via the Wi-Fi transceiver contain multiple actions. To separate each action so that each sample contains only one action process, this study employs a sliding window algorithm to partition the data.

In this study, the size of the sliding window and the step size L_0 for each slide are both set to 120. Thus, each time the window moves forward by one step, a sample of length L_0 is obtained. By performing multiple consecutive slides, a series of samples of length L_0 can be obtained. The dataset used was collected using a transmitter with one antenna and a receiver with three antennas, along with a modified Intel 5300 network card. The number of subcarriers was 30. After dividing each subcarrier using the sliding window algorithm and then integrating the results, the input data for the model was obtained. The input data has a dimension of 120×30, where 120 represents the step size and 30 represents the number of subcarriers.

TABLE I. OPERATION ENVIRONMENT CONFIGURATION

Equipment category	Equipment name
CPU	16 cores Intel(R) Xeon(R) Gold 6430
GPU	RTX 4090 (24GB)
Memory	120G
Python version	3.10.8
Pytorch version	2.1.2
CUDA version	12.1

TABLE II. MAIN PARAMETER SETTINGS

Parameter name	Parameter value
Epochs	50
Batchsize	128
Learning rate	1e-4
Optimizer	Adam

TABLE III. THE INFORMATION OF DATASET

Dataset	No. of actions	Type of actions
WiAR	10	High arm wave, Horizontal arm wave, Two hands wave, Draw x, Draw tick, High throw, Toss paper, Bend, Forward kick, Side kick
CSIAR	6	Stand up, Sit down, Lie down, Walk, Run, Fall,

V. Results and Analysis

The experiments in this section consist of two parts. The first part introduces PCNN, attention mechanism, and SSA parameter optimization based on the original WiTS model to conduct ablation experiments and analyze the results. The second part selects three commonly used models and compares them with the improved WiTS model to conduct comparative experiments and analyze the results.

A. Ablation Experiment

This research sequentially adds the corresponding improvement measures and conducts ablation experiments under the same experimental conditions to verify that each improvement introduced in this study enhances the performance of the original model in human action recognition. The results of these experiments are shown in Table IV.

Model A in the table represents the original WiTS model. Model B represents a model in which the convolutional structure of the CNN module is replaced with a partial convolutional structure and a spatial attention block is incorporated. Model C represents a model in which an attention mechanism is added to the LSTM module based on Model B. The last row represents the final model with SSA parameter optimization based on Model C, i.e., the model with all three improvements. The results show that all three improvement strategies have a positive effect on model performance, with varying degrees of improvement in accuracy across the WiAR and CSIAR datasets. Compared to the original model without any improvements, the accuracy is 93.84% and 94.57%, with parameters and Flops of 2.25M and 561.33M, respectively. For Model B, which replaced some convolutions, the accuracy improved slightly in both datasets despite a decrease of 0.06M in the number of parameters and 1.91M in Flops compared to the original model. Comparing Models B and C, the attention mechanism added in Model C does not significantly increase computational complexity, yet its accuracy improved by 0.62% and 1.26%, respectively, demonstrating the effectiveness of this improvement method. The final model achieved the best performance in ablation experiments, with accuracy reaching 95.75% and 96.71% on the two datasets, respectively, representing improvements of 1.91% and 2.14% over the unmodified model, and also showing a significant improvement over Model C. This further demonstrates that SSA can iteratively identify optimal hyperparameter configurations to enhance model performance.

TABLE IV. Results of Ablation Experiments

Model	PCNN	Attention mechanism	SSA	WiAR Acc./%	CSIAR Acc./%	Param. /M	Flops/M
A	×	×	×	93.84	94.57	2.25	561.33
B	√	×	×	93.86	94.60	2.19	559.42
C	√	√	×	94.48	95.86	2.19	559.42
Final	√	√	√	95.75	96.71	2.19	559.42

TABLE V. Results of Comparative Experiments

Dataset	Model	Precision/%	Recall/%	F1-score/%	Accuracy/%
WiAR	CNN	87.15	87.04	86.89	86.84
	TCN	86.82	85.60	85.89	85.14
	Bi-LSTM	80.88	78.37	78.49	78.13
	WiTS	**96.34**	**96.13**	**96.06**	**95.75**
CSIAR	CNN	87.26	86.11	86.10	86.57
	TCN	81.65	80.20	80.32	80.54
	Bi-LSTM	79.42	79.36	79.20	79.71
	WiTS	**97.13**	**96.52**	**96.39**	**96.71**

B. Comparative Experiment

To further validate the advancement of the proposed WiTS in human action recognition methods, this study selected three different deep learning models for comparison experiments. The experimental results are shown in Table V.

From the results, WiTS outperformed other models in all performance metrics in both experiments, while CNN ranked second in all results. Compared to CNN, in the WiAR dataset, the WiTS model improved Precision, Recall, F1-score, and Accuracy by 9.19%, 9.09%, 9.17%, and 8.91%, respectively; in the CSIAR dataset, these four metrics improved by 9.87%, 10.41%, 10.29%, and 10.14%, respectively, all showing significant improvements. In the experiments using the WiAR dataset, TCN ranked third in all metrics, closely following CNN and significantly outperforming Bi-LSTM. In another set of experiments, while TCN still ranked third, its performance metrics lagged significantly behind CNN, only slightly outperforming Bi-LSTM. The results indicate that a single network structure performs inconsistently across different datasets, making it challenging to accurately recognize human actions in real-world scenarios. In contrast, the WiTS model, which achieves deep integration of spatial and temporal features, demonstrates significantly superior performance compared to other models, fully validating the model's effectiveness.

VI. Conclusion and Future Work

A. Summary of Contributions

In the field of human action recognition using Wi-Fi CSI, the extraction of features from Wi-Fi signals using deep learning methods for the classification of actions has emerged as a prominent research direction in recent years. Nevertheless, these methods have yet to adequately extract information from the signal. The challenge of fully utilizing the information capabilities from both spatial and time dimensions to enhance model performance remains an urgent issue to be solved. To address this, this paper proposes a spatio-temporal hybrid neural network model named WiTS. This model integrates the advantages of CNN, TCN, Bi-LSTM, and Transformer. It uses CNN to extract spatial features, combines TCN and Bi-LSTM to achieve dual temporal dimension modeling, and incorporates the global attention mechanism of Transformer to comprehensively extract and multi-levelly fuse spatio-temporal features. Furthermore, this study optimizes and innovates the original WiTS model. The incorporation of partial convolution and spatial attention structures within the CNN module has been demonstrated to effectuate a reduction in model parameters and FLOPs, while preserving the model's capacity for feature extraction. Additionally, the attention mechanism is incorporated into the Bi-LSTM module to dynamically assign weight to temporal features, thereby enabling the model to prioritize key action segments. Finally, the SSA is employed to optimize hyperparameters, further enhancing model performance. Experiments on two complementary datasets, WiAR (primarily half-body actions) and CSIAR (primarily full-body actions), demonstrate that the model exhibits strong generalization capabilities across different action granularities, achieving average recognition accuracies of 95.75% and 96.71%, significantly outperforming single-network architectures. The F1-score exceeds 96% in both cases, addressing the issue of performance fluctuations in single-network models across different scenarios. Moreover, the enhanced model possesses a mere 2.19 million parameters and fewer than 560 million FLOPs offering advantages in terms of lightweight design and making it suitable for deployment on limited-computing edge devices while meeting real-time requirements.

B. Limitations and Future Work

Despite the WiTS model's demonstrated efficacy in action recognition accuracy and lightweight design, certain limitations persist. On the one hand, the experiments were validated exclusively on the WiAR and CSIAR public datasets, which exhibit limited diversity in action categories and scenarios. The exclusion of more complex scenarios, such as signal interference and varying room layouts, may impact the model's generalization capabilities in real-world settings. Secondly, the present research focuses primarily on the optimization of algorithms without addressing the practical deployment challenges that have been identified, such as dynamic environmental changes and long-term stability on edge devices.

To address these issues, the plan of future work encompasses three key areas. First, constructing larger, multi-scenario datasets incorporating more diverse human actions to enhance the trained model's generalization and robustness. Second, further optimizing the model architecture by exploring more adaptive learning mechanisms, enabling the model to dynamically adjust to environmental changes and user variations. Finally, deploy the WiTS model on real-world edge device platforms, conduct practical scenario testing, optimize inference efficiency and energy consumption, and advance its practical application in fields such as smart homes and health monitoring.

References

[1] Z. Shi, Q. Cheng, J. A. Zhang, and R. Y. Da Xu, "Environment-robust WiFi-based human activity recognition using enhanced CSI and deep learning," IEEE Internet of Things Journal, vol. 9, no. 24, pp. 24643–24654, 2022.

[2] Y. Zhang, F. Zhang, Y. Jin, Y. Cen, V. Voronin, and S. Wan, "Local correlation ensemble with GCN based on attention features for cross-domain person Re-ID," ACM Transactions on Multimedia Computing, Communications and Applications, vol. 19, no. 2, pp. 1–22, 2023.

[3] I. Ahmad, A. Ullah, and W. Choi, "WiFi-based human sensing with deep learning: Recent advances, challenges, and opportunities," IEEE Open Journal of the Communications Society, vol. 5, pp. 3595–3623, 2024.

[4] Z. Wei, W. Chen, S. Ning, W. Lin, N. Li, B. Lian, X. Sun, and J. Zhao, "A survey on WiFi-based human identification: Scenarios, challenges, and current solutions," ACM Transactions on Sensor Networks, vol. 21, no. 1, pp. 1–32, 2025.

[5] W. He, K. Wu, Y. Zou, and Z. Ming, "WiG: WiFi-based gesture recognition system," in 2015 24th International Conference on Computer Communication and Networks (ICCCN), Las Vegas, NV, USA, 2015, pp. 1–7.

[6] Y. Zhang, Y. Zheng, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Widar3.0: Zero-effort cross-domain gesture recognition with Wi-Fi," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 11, pp. 8671–8688, 2021.

[7] X. Meng, L. Feng, H. Chen, T. Chen, J. Ma, A. Wang, D. Liu, and Y. Zhao, "Just-in-time human gesture recognition using WiFi signals," Chinese Journal of Electronics, vol. 30, no. 6, pp. 1111–1119, 2021.

[8] X. Dang, Y. Liu, Z. Hao, X. Tang, and C. Shao, "Air gesture recognition using WLAN physical layer information," Wireless Communications and Mobile Computing, vol. 2020, no. 1, p. 8546237, 2020.

[9] Z. Hao, Y. Duan, X. Dang, Y. Liu, and D. Zhang, "Wi-SL: Contactless fine-grained gesture recognition uses channel state information," Sensors, vol. 20, no. 14, p. 4025, 2020.

[10] J. Huang, B. Liu, C. Miao, Y. Lu, Q. Zheng, Y. Wu, J. Liu, L. Su, and C. W. Chen, "PhaseAnti: An anti-interference WiFi-based activity recognition system using interference-independent phase component," IEEE Transactions on Mobile Computing, vol. 22, no. 5, pp. 2938–2954, 2021.

[11] A. Chelli, M. Muaaz, and M. Pätzold, "Actrec: a wi-fi-based human activity recognition system," in 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 2020, pp. 1–6.

[12] X. Cheng and B. Huang, "CSI-based human continuous activity recognition using gmm-hmm," IEEE Sensors Journal, vol. 22, no. 19, pp. 18709-18717, 2022.

[13] H. Zou, Y. Zhou, J. Yang, H. Jiang, L. Xie, and C. J. Spanos, "WiFi-enabled device-free gesture recognition for smart home automation," in 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 2018, pp. 476–481.

[14] M. Muaaz, A. Chelli, M. W. Gerdes, and M. Pätzold, "Wi-Sense: A passive human activity recognition system using Wi-Fi and convolutional neural network and its integration in health information systems," Annals of Telecommunications, vol. 77, no. 3, pp. 163–175, 2022.

[15] H. Yan, Y. Zhang, Y. Wang, and K. Xu, "WiAct: A passive WiFi-based human activity recognition system," IEEE Sensors Journal, vol. 20, no. 1, pp. 296–305, 2019.

[16] P. Duan, C. Li, J. Li, X. Chen, C. Wang, and E. Wang, "WISDOM: Wi-Fi-based contactless multiuser activity recognition," IEEE Internet of Things Journal, vol. 10, no. 2, pp. 1876–1886, 2022.

[17] W. Meng, X. Chen, W. Cui, and J. Guo, "WiHGR: A robust WiFi-based human gesture recognition system via sparse recovery and modified attention-based BGRU," IEEE Internet of Things Journal, vol. 9, no. 12, pp. 10272–10282, 2021.

[18] R. Gao, W. Li, J. Liu, S. Dai, M. Zhang, L. Wang, and D. Zhang, "WiCGesture: Meta-motion-based continuous gesture recognition with Wi-Fi," IEEE Internet of Things Journal, vol. 11, no. 9, pp. 15087–15099, 2023.

[19] W. Cui, B. Li, L. Zhang, and Z. Chen, "Device-free single-user activity recognition using diversified deep ensemble learning," Applied Soft Computing, vol. 102, p. 107066, 2021.

[20] J. Chen, S. H. Kao, H. He, W. Zhuo, S. Wen, C. H. Lee, and S. H. G. Chan, "Run, don't walk: chasing higher FLOPS for faster neural networks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, Canada, 2023, pp. 12021–12031.

[21] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," Systems Science & Control Engineering, vol. 8, no. 1, pp. 22–34, 2020.

[22] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," Journal of Big Data, vol. 8, no. 1, p. 53, 2021.

[23] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in European conference on computer vision, D. Fleet, T. Pajdla, B. Schiele and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.

[24] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," arXiv preprint arXiv:1803.01271, 2018.

[25] J. Fan, K. Zhang, Y. Huang, Y. Zhu, and B. Chen, "Parallel spatio-temporal attention-based TCN for multivariate time series prediction," Neural Computing and Applications, vol. 35, no. 18, pp. 13109–13118, 2023.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 2016.

[27] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez and Ł. Kaiser, "Attention is all you need," in Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach California USA, 2017, pp. 6000–6010.

[29] T. Lin, Y. Wang, X. Liu, and X. Qiu, "A survey of transformers," AI Open, vol. 3, pp. 111–132, 2022.

[30] S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaee, "A survey on behavior recognition using WiFi channel state information," IEEE Communications Magazine, vol. 55, no. 10, pp. 98–104, 2017.

[31] L. Guo, L. Wang, C. Lin, J. Liu, B. Lu, J. Fang, Z. Liu, Z. Shan, J. Yang, and S. Guo, "Wiar: A public dataset for WiFi-based activity recognition," IEEE Access, vol. 7, pp. 154935–154945, 2019.

Automated Scoliosis Diagnosis in Spinal Imaging: Laboratory Validation, Clinical Limitations, and Systematic Implementation Challenge Review

Ervin Gubin Moung[1], Xie Aishu[2], Ali Farzamnia[3]
Faculty of Computing and Informatics, Universiti Malaysia Sabah, Jalan UMS, Kota Kinabalu, 88400, Sabah, Malaysia [1, 2]
School of Computing and Engineering, University of Huddersfeld, West Yorkshire, HD1 3DH, United Kingdom [3]

***Abstract*—Technological advances in automated medical imaging diagnosis have created translation gaps between laboratory achievements and clinical implementation, with traditional manual Cobb angle measurement requiring considerable time with inevitable measurement errors. This review analyzes translation challenges in automated diagnosis systems using scoliosis assessment as a case study, examining 55 articles from 1948-2025 across three domains: Cobb angle measurement, classification, and segmentation. Despite research investment, fully automated approaches have not surpassed semi-automated performance in comparable validation studies. Within the 23 Cobb angle measurement studies, traditional methods outperform sophisticated deep learning systems with average error rates of 1.8° ± 0.4° MAD versus 4.2° ± 1.8° MAE, while validation degradation occurs with performance dropping from 95.28% to 85.9% when transitioning to real-world datasets. Non-standard classification achieves high accuracy but lacks clinical utility, while standard systems struggle with automation, revealing a translation paradox where technical sophistication does not correlate with clinical adoptability. Main problems include testing method gaps, performance drops, different automation approaches, and cost issues. This review recommends standard testing methods and step-by-step clinical implementation to help these systems work in real clinics.**

***Keywords*—Automated diagnosis; medical imaging; scoliosis; Cobb angle; clinical implementation; artificial intelligence**

I. Introduction

Medical imaging has undergone major advances over the past seven decades, with automated systems achieving high performance in laboratory settings. Despite these achievements, translation into clinical practice remains limited. Scoliosis assessment illustrates this gap clearly. While AI systems for diabetic retinopathy and breast cancer detection have achieved clinical success with standardized protocols and FDA approval [IDx-DR, 100,000 patients annually; breast cancer AI 94.5% sensitivity], scoliosis measurement systems remain confined to research laboratories.

The field has experienced continuous development of diagnostic approaches since 1948, yet clinical challenges remain unresolved. Researchers have proposed multiple measurement principles for scoliosis assessment, including the Ferguson method [1], Greenspan method [2], Diab method [3], and Centroid method [4]. These developments demonstrate ongoing efforts to enhance diagnostic accuracy and reliability. The Cobb angle measurement method was developed by Cobb in 1948 and adopted by the Scoliosis Research Society as the standard method for quantifying scoliotic deformities in 1966. This method remains the clinical gold standard despite well-documented limitations. The traditional manual measurement process has problems with errors between different doctors and even the same doctor measuring twice [5], with studies showing significant differences in measurements [6][7][8][9]. The Ferguson method exhibits considerable intra-observer and inter-observer variation [1]. Measurement inconsistencies continue to exist across established methodologies [10][11][12]. Since 2017, focus has transferred to estimating Cobb angles using Artificial Intelligence (AI) models, with researchers [13] implementing computer-aided measurement based on automatic detection of vertebral slopes using deep neural network, and [14] proposing a Convolutional Neural Network (CNN)-based spine and Cobb angle estimator using moire images. Modern approaches like [15] achieved impressive results using convolutional neural networks for Cobb angle measurement, while [16] achieved high performance using U-Net webserver for spine segmentation, and [17] demonstrated effective lumbar vertebrae segmentation. This technological sophistication has not translated proportionally into clinical implementation rates, revealing limitations that distinguish scoliosis AI from successful medical AI implementations in radiology and pathology.

A. Scope and Objectives

To address this problem, this review asks three main questions: (1) What factors prevent laboratory-validated systems from achieving comparable performance in clinical settings? (2) What barriers hinder adoption of these systems in actual workflows? and (3) What strategies can bridge the gap between laboratory development and clinical implementation?

B. Paper Organization

The remainder of this paper is organized as follows: the Methodology section outlines the literature search and study selection. Background and Context review historical development and contrasts scoliosis AI with successful medical AI domains. Current State Analysis summarizes recent performance trends, while Critical Analysis investigates the causes of translation failures. Future Directions provide recommendations, and Conclusions synthesize key findings for stakeholders.

II. Methodology

A. Search Strategy and Study Selection

A comprehensive literature search was conducted across multiple databases including PubMed/MEDLINE, IEEE Xplore, ACM Digital Library, and Scopus from January 1948 to February 2025. The search strategy combined controlled vocabulary terms and keywords related to scoliosis, automated measurement, artificial intelligence, and clinical implementation. Primary search terms included: ("scoliosis" OR "spinal curvature") AND ("automated" OR "computer-assisted" OR "artificial intelligence" OR "deep learning" OR "machine learning") AND ("Cobb angle" OR "measurement" OR "classification" OR "segmentation"). Additional studies were identified through citation tracking of seminal papers and review of reference lists.

B. Inclusion and Exclusion Criteria

Inclusion criteria were: (1) English-language, peer-reviewed articles or conference proceedings; (2) studies on automated or computer-assisted scoliosis assessment; (3) work covering Cobb angle measurement, scoliosis classification, or spinal segmentation; (4) studies reporting quantitative performance metrics; and (5) articles providing enough methodological detail for analysis. Exclusion criteria were: (1) research focused only on treatment or surgical interventions; (2) articles lacking performance metrics or validation data; (3) duplicate publications, including conference papers later published as journal articles; (4) studies on non-human subjects; and (5) purely theoretical work without experimental validation.

C. Study Categorization and Data Extraction

Selected studies were categorized into three primary domains: (1) Cobb angle measurement approaches, (2) scoliosis classification systems, and (3) spinal structure segmentation methods. This categorization was based on the primary research objective and methodology reported in each study. Data extraction followed a standardized protocol that captured: study characteristics (authors, year, study design, sample size), technical approaches (algorithms employed, imaging modalities), performance metrics (accuracy measures, reliability statistics, validation methods), and clinical implementation details (automation level, workflow integration, clinical testing). Performance metrics were standardized where possible, with accuracy measures converted to comparable units (degrees for Cobb angle measurement, percentages for classification and segmentation). When studies reported multiple metrics, priority was given to the most clinically relevant measure (e.g., mean absolute difference for Cobb angles, overall accuracy for classification). The selection process for this review followed established guidelines for literature identification and screening. This review analyzed 55 studies spanning from 1948 to 2025 across the three primary domains of automated scoliosis assessment. Fig. 1 illustrates the comprehensive search and selection methodology employed to identify the final 55 studies included in this review, demonstrating the distribution across the three primary domains of automated scoliosis assessment.

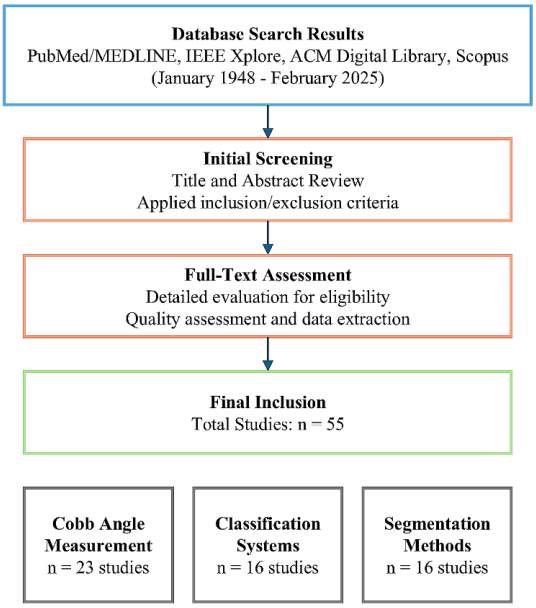


Fig. 1. Study selection flow diagram for review of automated scoliosis.

The flowchart shown in Fig. 1 demonstrates the systematic selection process from initial database searches through final inclusion, resulting in 55 studies categorized across three primary domains: Cobb angle measurement (n=23), classification systems (n=16), and segmentation methods (n=16). The distribution of studies across the three domains reflects the evolution of research focus over the 77-year timeframe examined. Cobb angle measurement studies represent the largest portion (42% of total), reflecting the central importance of angular measurement in scoliosis assessment. Classification and segmentation approaches each represent 29% of studies, indicating that researchers recognize effective automation requires progress across multiple complementary technologies. This balanced distribution provides comprehensive coverage of translation challenges across all aspects of automated scoliosis assessment.

D. Limitations

This review has several limitations: (1) differences in performance reporting across studies limited quantitative analysis; (2) publication bias may favor positive results, potentially overestimating automated system performance; (3) limited availability of real-world clinical validation data restricted translation analysis; (4) rapid technological changes may make some analyses outdated; (5) focus on English-language publications may have excluded relevant international research.

III. Background and Context

A. Evolution of Automated Scoliosis Assessment

The field has evolved through four technological phases, each with different strengths, weaknesses, and adoption rates. As technological sophistication increased, clinical implementation declined. Table I presents the development phases and their duration, showing rapid technological development but declining clinical implementation success.

TABLE I. TECHNOLOGY DEVELOPMENT PHASES

Phase	Period	Duration
Manual Era	1948-1990	42 years
Early Automation	1990-2010	20 years
Classical ML Peak	2010-2017	7 years
Deep Learning Era	2017-Present	7+ years

The rapid development in Table I differs from the gradual, steady progress seen in successful medical AI domains like mammography, where evolution occurred over similar timeframes but achieved progressive clinical integration rather than decreased adoption. The performance and clinical implementation patterns across these phases show the main problem. Table II shows that as technical performance improved, clinical implementation decreased.

TABLE II. PERFORMANCE VS. CLINICAL IMPLEMENTATION

Phase	Best Performance	Clinical Implementation
Manual Era	Intra: 5.6°, Inter: 6.6° [1]	100%
Early Automation	Variable improvement [18][11][12]	<5%
Classical ML Peak	MAD: 1.5°, ICC: >0.95 [10][19]	<5%
Deep Learning Era	MAE: 4.9°, DC: 95.8% [20][21]	<1%

Table II shows a pattern that differs from successful medical AI domains. In successful domains, better technology leads to more clinical use. Diabetic retinopathy screening moved from research to FDA approval to widespread use over a similar timeframe. It was found that specific factors in scoliosis context, rather than general AI problems, cause implementation difficulties, which suggest that targeted approaches may perform better and show improved progress within clinical practice settings.

B. *Medical AI Translation Context: Lessons from Successful Domains*

Scoliosis AI faces unique translation challenges. Comparison with successful medical AI implementations reveal differences in problem structure and validation approaches, which suggest that targeted solutions perform better within scoliosis treatment contexts.

1) Successful medical AI implementations: It was found that diabetic retinopathy screening achieved clinical success mainly because the task was simplified into binary classification of present or absent, with the added benefit of standardized imaging protocols that allowed for consistent practice within clinical workflows. Clear clinical decision points together with regulatory pathways appear to support adoption. Breast cancer detection AI perform better by supporting radiologists, which mean it show improved progress in accuracy without full automation. These cases suggest clinical acceptance is more likely when AI act as enhancement with professionals, not as substitutes.

2) Scoliosis AI differences: Scoliosis assessment was found to require continuous measurement instead of binary classification, and this make the task more complex than other screening-based AI applications. It lack standardized imaging protocols across hospitals, which suggest that consistent practice remain difficult to achieve. Scoliosis assessment also need to integrate with complex treatment decision algorithms, and this requirement only perform better when AI is used as enhancement rather than full replacement. In addition to that, the current regulatory frameworks lack provisions for automated measurement systems, so the approval pathway remain unclear. These fundamental differences, repeated again to emphasize, clarify why proven strategies from other medical AI applications prove ineffective when directly applied to scoliosis assessment, and this suggest improvement will only be achieved by adopting different design approaches. Comparing successful and failed medical AI implementations shows major differences in problem structure, implementation strategies, and clinical outcomes. Fig. 2 illustrates the key distinguishing factors between successful medical AI domains and the failed scoliosis AI domain, highlighting the strategic and technical decisions that determine translation success.

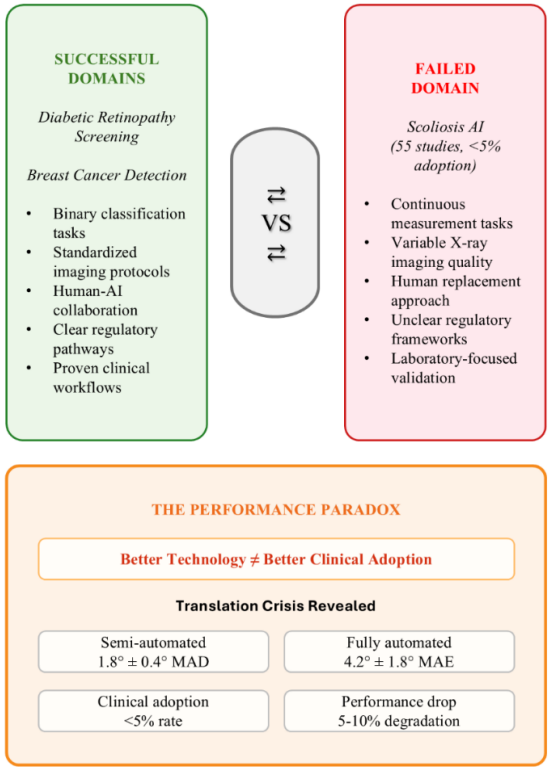


Fig. 2. Medical AI success vs failure analysis.

Fig. 2 shows how important it is to match the right technology with the right problem for medical AI success. Successful domains benefit from binary classification tasks that align with deep learning strengths, while scoliosis AI requires continuous geometric measurement that favors traditional mathematical approaches. The performance paradox contradicts conventional expectations. Semi-automated methods achieve better results (1.8° ± 0.4° MAD) compared to fully automated systems (4.2° ± 1.8° MAE). This finding shows that current AI approaches do not match the geometric precision needed for scoliosis assessment. This analysis helps

explain why helping doctors work better than replacing doctors for successful clinical use. The foundation era established measurement principles through the Ferguson method [1], Greenspan method [2], and Diab method [3], but these were designed for manual implementation. Early computer-assisted approaches [18][11][12] showed promise but remained time-consuming due to manual endpoint identification requirements. In [18], authors compared radiographic and computer-assisted measurements, while in [11] and [12], the authors evaluated manual versus computer-assisted approaches. Traditional image processing and classical machine learning approaches achieved peak performance during 2010-2017. In [19], the authors achieved ICC>0.95 using Fuzzy Hough Transform [22] and [23] achieved ICC>0.9 using Active Shape Models. However, these successes occurred during a period when other medical AI domains were transitioning from research to clinical implementation. This timing highlights scoliosis AI's unique translation challenges.

C. Deep Learning Era and Divergence from Medical AI Success Patterns

Deep learning approaches led to a shift toward complete automation instead of human-AI collaboration that worked well in other medical domains. This divergence explains much of the current translation crisis.

Several researchers developed deep learning solutions for scoliosis assessment. In research by [13], authors developed computer-aided Cobb measurement using deep neural networks, while researchers in [14] proposed CNN-based estimators and those in [15] achieved significant results using convolutional neural networks.

Advanced segmentation approaches also emerged. In studies by [16], researchers deployed U-Net achieving 90.4% dice coefficient, and authors in [24] proposed comprehensive spine detection frameworks. In research by [25], authors developed automatic vertebrae localization for CT scans, researchers in [28] demonstrated CNN-based lumbar spine segmentation, and those in [29] contributed deep learning methods for spine analysis in MR images. Other technical approaches included [30] implementing BoostNet for automatic landmark estimation, [31] developing structured multi-output regression for direct Cobb angle estimation, and [32] proposing vertebra detection and corner regression techniques. However, this period showed a different pattern from successful medical AI domains. While domains like radiology succeeded through augmentation approaches that enhanced radiologist capability, scoliosis AI pursued replacement approaches that attempted complete automation. Classification approaches showed similar patterns, with [33] achieving results through local centroids evaluation, [34] implementing support vector classifiers for progression risk prediction, and [35] developing machine learning approaches for clinical classification using 3D surface data. Feature-based approaches like [36] demonstrated vertebrae localization and segmentation for curvature classification, while [37] applied support vector machines for severity assessment from surface topography. Recent developments include [38] analyzing scoliosis from spinal X-ray images, [39] implementing artificial neural networks for spinal deformity identification, and various classification frameworks [40][41][42][43] addressing different aspects of scoliosis assessment with varying degrees of automation success. Clinical reality involves complex decision-making beyond just measurement accuracy. The Lenke classification system [44] serves as the popular standard, but it requires automation of the entire Cobb method instead of providing decision support for specific components. Studies evaluating classification reliability [45] demonstrate the complexity of implementing automated systems that match clinical standards.

D. Summary

The analysis shows that scoliosis AI fails not because AI itself is bad, but because researchers chose the wrong approach compared to successful AI in other medical areas. Understanding these basic differences is important for analyzing current performance patterns and finding effective solutions. The next section examines how these different approaches affect current system performance and clinical implementation. Building on this analysis, the following section examines current performance patterns across the three primary domains.

IV. Current State Analysis

A. Cobb Angle Measurement Approaches

Automated Cobb angle measurement demonstrates a concerning trend where increased technological sophistication correlates with decreased clinical adoption. This paradox becomes more pronounced when compared to successful medical AI domains where sophistication enhances rather than hinders adoption. Analysis of 23 studies identified three distinct methodological approaches: traditional methods (3 studies), semi-automated approaches (8 studies), and fully automated systems (12 studies). This contradicts patterns in successful medical AI domains where deep learning typically outperforms traditional approaches, suggesting differences in problem structure. Table III presents an analysis of performance across different methodological approaches, including Mean Absolute Difference (MAD) and Mean Absolute Error (MAE) where available from the literature. Performance metrics vary across studies due to different datasets, validation methods, and measurement protocols. Direct numerical comparison should be interpreted cautiously due to methodological differences in error calculation across studies.

TABLE III. Average Performance Analysis Across Study Types

Method Type	Performance	Studies (n)	Translation Rate
Manual	3.9° ± 0.3° MAD	Baseline	100% (standard)
Semi-Automated	1.8° ± 0.4° MAD	8	<5% adoption
Fully Automated	4.2° ± 1.8° MAE	12	<1% adoption

Table III shows performance metrics as mean ± standard deviation across multiple studies on semi-automated and fully automated approaches, revealing statistically significant performance differences that contradict expectations based on successful medical AI domains.

Semi-automated methods achieve better performance than deep learning systems (1.8° ± 0.4° MAD vs 4.2° ± 1.8° MAE)

for three primary reasons. First, continuous measurement tasks favor geometric approaches over pattern recognition (Task-Method Mismatch). Second, deep learning models optimize for controlled datasets that poorly represent clinical variability (Overfitting to Laboratory Conditions). Third, end-to-end learning discards domain-specific geometric constraints that are essential for reliable measurement (Feature Engineering Loss). Traditional approaches achieved superior performance, with researchers in [19] reaching ICC>0.95 using Fuzzy Hough Transform [22] because geometric constraints built into spine anatomy provide more reliable guidance than learned features. Semi-automated approaches like [46] achieved significant accuracy using generalized Hough transform for cervical vertebrae detection. Statistical modeling approaches [47] demonstrated effective lumbar spine segmentation using multi-vertebrae anatomical shape models. Modern approaches struggle: [20] achieved MAE of 4.9º and [32] achieved SMAPE of 25.69% because they try to learn geometric relationships that work better when expressed through direct mathematical formulations. Advanced segmentation approaches [48] using iterative fully convolutional neural networks showed promise but faced translation challenges.

B. Classification Systems Analysis

Scoliosis classification analysis shows a key difference that separates this domain from successful medical AI implementations. While successful domains like pathology AI achieve clinical utility through high-performing automated classification, scoliosis classification faces a utility-automation trade-off that prevents clinical implementation. The analysis shows that automation success decreases as clinical utility increases, creating an implementation challenge unique to complex measurement-dependent medical domains. Table IV presents the classification performance characteristics across different system types.

TABLE IV. CLASSIFICATION TRADE-OFF ANALYSIS

Classification Type	Automation Success	Clinical Utility
Non-Standard Binary	High (88.3%) [34]	Limited
Non-Standard Ternary	High (94.69%) [37]	Limited
Standard (Lenke)	Low (72%) [43]	High

This utility-automation trade-off occurs because clinically relevant classification systems like Lenke [44] require multiple geometric and morphological features that are difficult to automate. Automated classification only succeeds when simplified to features that reduce clinical relevance. This differs fundamentally from successful medical AI domains where clinical relevance and automation potential align.

Non-standard approaches show better automation performance through simplified feature sets. Authors in [33] achieved 88.3% accuracy using local centroids evaluation because geometric simplification enables reliable automation, while researchers in [36] achieved 94.69% accuracy through feature-based approaches that sacrifice comprehensive clinical assessment for automation reliability. Support vector approaches [34][37] achieved different levels of success based on feature complexity and clinical requirements. Recent classification developments include [38] analyzing scoliosis from spinal X-ray images, [39] implementing total curvature analysis with artificial neural networks, and [43] developing segmentation network-based Lenke classification systems. However, standard classification systems face dependency on reliable Cobb angle automation, creating cascading failures where measurement errors spread through classification algorithms. Researchers in [40] achieved 92.9% accuracy using Decision trees, and those in [41] achieved kappa values of 0.94 through rule-based approaches, but these successes depend on manual measurement inputs rather than automated systems. Authors in [42] demonstrated novel classification methods using 3D ultrasound imaging, while reliability studies [45] highlight the challenges of automated implementation of established classification systems.

C. Segmentation Technologies and Validation Crisis

Spinal structure segmentation represents the most critical component for understanding validation failures that distinguish scoliosis AI from successful medical AI domains. Analysis of 55 articles demonstrates impressive laboratory achievements but shows real-world performance degradation patterns not observed in successful medical AI implementations. The consistent performance degradation stems from differences between scoliosis imaging characteristics and successful medical AI domains. Scoliosis assessment relies on X-ray imaging that carry variability in positioning, exposure, and anatomical presentation. This differs from the standardized imaging protocols that exist in diabetic retinopathy screening or mammography. Laboratory validation often fails to capture such clinical variability, which suggests that results observed in controlled settings do not reflect actual practice. Table V presents the validation crisis evidence across different segmentation approaches, and it demonstrates degradation patterns that appear unique to this domain.

TABLE V. VALIDATION CRISIS EVIDENCE

Approach	Lab Performance	Real-World Drop
Traditional (ASM)	93.6% [17]	5-8% typical
U-Net Variants	95.8% [21]	10% [50]
Instance Segmentation	96% [51]	7% [52]

Table V shows that performance drop mainly come from three causes. First is Dataset Bias Amplification, where laboratory datasets usually do not include the wide imaging variation that is often found in real clinical settings. Second is Anatomical Anomaly Underrepresentation, where the training data has too few cases of complex deformity and this makes generalization weaker. Third is Workflow Integration Failure, since experimental testing often ignores positioning and the image quality limitation that exist in daily practice. It was found that traditional approaches perform better and maintain more stability. For example, landmark-based segmentation [17] reports 93.6% accuracy, while the generalized Hough transform [46] also shows strong outcome. These methods stay more consistent because geometric constraints can handle imaging variation better. In comparison, statistical [47] and iterative methods [48] produce mixed stability. At some time, they perform better, but other times they become weaker. Advanced deep learning approaches suffer sharper drops. RAR-U-Net [21] with reported 95.8% accuracy degraded

strongly under clinical-like testing. LPAQR [49] at 95.28% also degrade since the learned features do not transfer well when imaging condition changes. Newer studies bring in region-based CNNs [50], attention gate dual-pathway networks [51], detection-guided mixed-supervised segmentation [52], and automated vertebrae recognition [53]. Yet the validation issue still remains. This situation stands in clear contrast with other medical AI fields, where real-world results usually match or even surpass laboratory results. It was found that the main difference comes from dataset representation and workflow integration during system development. Whole spine segmentation [54] tries to address this gap, but translation barriers still not fully resolved.

This validation gap stands in sharp contrast with other medical AI fields that usually observe better real-world outcomes, where performance in clinical settings often match or even surpass laboratory benchmarks. It was found that the key difference lies in dataset representation and workflow integration during system development. Whole spine segmentation methods [54] attempt to respond to this challenge, but translation barriers still remain unresolved. Early computer-assisted systems [55] already establish the idea that digital methods can improve measurement consistency, though they still require much manual intervention.

D. Comparative Performance Analysis

Table VI presents comparative advantages of different approaches across scoliosis assessment and successful medical AI domains.

TABLE VI. IMPLEMENTATION PRIORITY MATRIX

Approach	Performance	Clinical Stability	Translation Rate	Key Advantage
Scoliosis - Semi-automated	1.8° ± 0.4° MAD	High	<5%	Geometric precision
Scoliosis - Fully automated	4.2° ± 1.8° MAE	5-10% degradation	<1%	Pattern learning
Manual (Baseline)	5.6°/6.6° variation	100% stable	100%	Human expertise
Diabetic retinopathy AI	94.5% sensitivity	Stable	>80%	Binary classification
Breast cancer AI	High accuracy	Stable	>60%	Augmentation strategy

Semi-automated methods offer clear precision benefit while also achieving clinical stability that looks similar to other successful medical AI applications. Mathematical approaches with geometric constraints [19][22] work better for scoliosis assessment compared to deep learning methods, which mainly excel in pattern recognition tasks. Measurement standardization provides unique advantage that is not available in diagnostic areas requiring subjective clinical judgment. Hybrid augmentation frameworks follow proven medical AI strategies and at the same time meet the specific demand of geometric measurement tasks. This combined approach delivers better reliability by merging accuracy with quality improvement, and it creates benefit that neither automation nor manual methods achieve alone.

E. Summary

These performance differences highlight gap between scoliosis AI and proven medical AI systems. The next section explores why standard methods struggle in this field. It also looks at what this tells us about requirement essential for clinical deployment in complex measurement tasks.

V. CRITICAL ANALYSIS: THE TRANSLATION CHALLENGE

A. Clinical Implementation Barriers for Scoliosis AI

Scoliosis AI struggles in clinical practice for reasons different from other medical AI applications. Understanding these barriers requires examining why standard methods break down in measurement tasks that demand geometric precision instead of only pattern recognition.

1) Primary barrier: *Task-Technology Mismatch:* The core implementation failure stem from mismatch between deep learning optimization goals and the geometric measurement need. This suggest improvement require redirecting effort from optimization aim toward accuracy of measurement that perform better for clinical use. Successful medical AI fields like image classification leverage pattern recognition strength of deep learning. Cobb angle measurement demand geometric accuracy that mathematical methods deliver more consistent.

2) Secondary barrier: *Validation Approach Differences:* Scoliosis AI validation method diverge from proven medical AI practice by focusing on laboratory performance score rather than workflow compatibility. Systems tested only in controlled setting fail to show improved progress for clinical use, while implementation in real hospital environment perform better and observe better alignment with workflow.

3) Tertiary barrier: *Regulatory Framework Gaps:* Other medical AI field perform better because they benefit from clear regulatory pathway for screening and diagnostic support. Scoliosis measurement automation operate without such structure, so approval obstacle limit adoption and prevent system from showing improved progress toward commercial development. Table VII analyze the factor that explain why scoliosis AI fail to observe better clinical integration across system aspect.

TABLE VII. TRANSLATION FAILURE MECHANISMS

Failure Dimension	Primary Cause	Success Comparison
Technical Performance	Task-Technology Mismatch	Radiology (pattern recognition)
Validation Crisis	Laboratory-Clinical Gap	Pathology (workflow integration)
Clinical Utility	Automation-Utility Trade-off	Diagnostic AI (augmentation)

Examining why implementations fail reveals that applying proven medical AI approach in unsuitable areas leads to consistent breakdown. Technical performance issue appears because deep learning performs better in pattern recognition task like diabetic retinopathy detection but struggle in

geometric measurement task that require mathematical precision. The validation crisis emerges because controlled laboratory conditions fail to represent imaging variability that exists in clinical scoliosis assessment. This contrasts with standardized imaging protocol used in successful domain, where practice observes better consistency. Successful medical AI implementation shares common characteristics that scoliosis AI lack standardized imaging protocol, binary or categorical output, augmentation instead of replacement strategy, and clear regulatory pathway. The root cause of translation failure comes from pursuing full automation, while clinical adoption would show improved progress if human-AI collaboration performs better in practice.

B. Main Problems and Cross-Domain Comparison

The challenges preventing scoliosis AI from achieving clinical implementation differ fundamentally from the enabling factors in successful medical AI domains, indicating that scoliosis requires domain-specific solutions for clinical adoption.

1) Validation crisis analysis: The validation crisis results from structural differences between scoliosis AI and successful medical AI domains [26] [27]. Performance dropped in multiple studies, with [49] reporting a decline from 95.28% to 85.9%, [51] from 95.4% to 89%, [52] from 94.39% to 87.21%, and [53] from 95.19% to 93.89%. This pattern contrasts with successful medical AI domains where real-world performance typically meets or exceeds laboratory performance. Recent validation studies continue to confirm this pattern, with [56] reporting similar reliability challenges in mobile AI applications, and [57] demonstrating that even advanced CNN approaches maintain this performance gap between laboratory and clinical settings.

2) Understanding validation failures: Laboratory validation employs cross-validation techniques on limited datasets that fail to capture clinical variability. Successful medical AI domains use standardized imaging protocols. Scoliosis assessment involves variable positioning, exposure settings, and anatomical presentations. Laboratory datasets systematically exclude this clinical variability. This creates optimistic performance estimates that fail to predict clinical utility.

3) Cross-domain economic analysis: Economic challenge in scoliosis AI contrast strongly with successful application. Diabetic retinopathy screening show improved progress by delivering clear cost benefit through early detection and fewer specialist visits. In comparison, scoliosis measurement automation face setup cost that outweigh workflow gain. Manual measurement take about 20 minutes [10], yet automated system require heavy infrastructure investment that fail to perform better in workflow advantage.

4) Regulatory framework differences: Regulatory obstacle appear because no suitable framework exist for measurement automation. Other medical AI field perform better because they use existing approval pathway for diagnostic support and screening tool. Scoliosis automation, in contrast, need new regulatory structure that balance accuracy requirement with clinical safety.

C. Economic and Implementation Reality

Economic analysis observes better clarity when comparing scoliosis AI with successful field. The difference lies in cost structure, setup need, and value clarity.

1) Cost-benefit structure analysis: Real-world implementation show automation expense often outweigh saving once development, validation, training, maintenance, and regulatory cost are included. Successful medical AI field show improved progress because they provide clear value benefit. Diabetic retinopathy screening perform better by reducing specialist visits and allowing earlier treatment, while scoliosis automation show vague benefit compared to setup cost.

2) Infrastructure requirements: X-ray images suffer from poor quality and weak contrast, and this limit automated system performance. Alternative imaging such as CT and MR create high expense and radiation risk. Because of these constraint, X-ray remain the preferred choice due to cost, radiation, dataset availability, and maturity. Yet this preference not show improved progress for automation, since it create obstacle that prevent system from performing better.

3) Clinical integration complexity: Integration barriers go beyond technical performance to include workflow compatibility challenges, regulatory compliance demands, and organizational adaptation requirements. Major integration obstacles include workflow disruption needing staff retraining, system reliability standards exceeding current abilities, complicated error management for system breakdowns, and new quality control procedures for combined workflows.

4) Regulatory framework inadequacy: Existing classification systems like Lenke [44] lack development in fully automated applications, creating regulatory challenges where standard validation methods cannot properly evaluate AI systems. Regulatory obstacles include unclear FDA approval routes for AI measurement systems, responsibility assignment issues for automated errors, undefined quality benchmarks for AI medical measurements, and unspecified monitoring requirements for developing automated systems.

D. Summary

It was found that recognizing these barriers can create groundwork for building effective solutions that draw from proven medical AI approaches, and in terms of measurement this becomes very important for geometric tasks. Moreover, the following section will explain some strategic approaches that follow established success model in medical AI, while also trying to answer the particular challenge faced in automated scoliosis assessment.

VI. Discussion

This review of 70 years of scoliosis AI research shows that common belief in medical AI, where better technology directly creates clinical use, does not hold here. In fact, deployment

depends more on strategic fit than on complexity alone. It was found that increasing complexity often reduces adoption rate, opposite to what seen in other successful medical AI fields, and this suggests scoliosis AI face structural issues rather than only simple implementation barrier. The difficulty is less about accuracy itself, but more about how the system can fit into clinical workflow.

The core mismatch between geometric measurement requirement and pattern recognition approach highlights a wider habit in medical AI: applying proven technique everywhere without enough task-specific adjustment. This study indicates that problem features, and not only algorithm performance, is what truly shape clinical success. Moreover, the consistent result that semi-automated methods perform better than fully automated systems suggest that human and AI partnership may actually be more effective path forward. In terms of accuracy this is very important, because certain areas need precise measurement rather than just pattern recognition, and this again confirms that the task requirement is more critical than algorithm complexity.

The validation crisis observed across multiple studies indicates systematic methodological inadequacies in how scoliosis AI research approaches real-world applicability. Unlike successful medical AI domains that prioritize clinical workflow integration during development, scoliosis AI research appears to treat clinical implementation as a secondary concern addressed after algorithmic optimization. This finding aligns with broader medical AI translation challenges identified by [58], who demonstrate that healthcare AI failures stem primarily from inadequate attention to clinical implementation factors during development, rather than technical limitations. This sequence reversal may explain why laboratory performance fails to predict clinical utility consistently.

Several limitations affect these findings. The focus on English-language publications may have excluded relevant international developments, particularly given the global nature of scoliosis research. The rapid pace of technological change means some recent advances may not yet demonstrate their clinical translation potential, and the 77-year timeframe may obscure recent improvement trends. Furthermore, viewing outcomes as simply success or failure may oversimplify the complex factors affecting clinical adoption, where partial implementation or hybrid workflows could represent significant progress that traditional metrics miss.

These insights extend beyond scoliosis to other measurement-based medical fields. Areas needing geometric accuracy, continuous measurement, or integration with established mathematical methods may work better with hybrid approaches than complete automation. This indicates the medical AI field could benefit from developing task classifications that systematically match technological methods to problem features instead of assuming successful techniques work universally.

Economic review shows implementation expenses frequently outweigh potential benefits when including system development, validation, training, maintenance, and regulatory costs. This challenges the belief that better technical performance automatically creates strong value propositions. Healthcare organizations seem to make logical economic choices based on realistic cost-benefit evaluations, suggesting that apparent resistance to implementation may reflect sound economic assessment. Recent comprehensive analysis of AI implementation obstacles [59] confirms economic factors as the main barrier to healthcare AI adoption, supporting the logical decision-making approach seen in scoliosis AI.

This review suggests medical AI research should consider clinical implementation factors earlier in algorithm development, rather than treating real-world application as an afterthought. The scoliosis AI experience offers important lessons for emerging medical AI fields about the value of strategic alignment, appropriate technology choices, and realistic validation approaches in achieving genuine clinical impact.

VII. Future Directions and Recommendations

A. Strategic Research Priorities Based on Current Findings

Future research needs to tackle the core challenges found in this study while applying lessons from successful medical AI applications in other fields. The top priority involves moving from complete automation strategies to enhancement approaches that have worked well in radiology and pathology AI systems.

1) Mechanism-Based research priorities: Our analysis shows research priorities must solve the task-technology mismatch by creating hybrid approaches that blend geometric mathematical methods with AI enhancement instead of full replacement. This matches successful medical AI patterns where technology supports rather than replaces human expertise. Recent progress in automated Cobb angle measurement [60], [61] shows that hardware-independent approaches can reach clinical accuracy when designed as enhancement rather than replacement systems.

2) Learning from successful medical AI domains: Diabetic retinopathy screening succeeded through standardized procedures, binary decision outputs, and clear regulatory routes. Breast cancer detection AI gained adoption through enhancement strategies that improved rather than replaced radiologist skills. Scoliosis AI must follow similar enhancement approaches, emphasizing measurement quality improvement and error reduction instead of complete automation. Table VIII shows the strategic implementation plan based on our analysis and successful medical AI patterns.

TABLE VIII. Implementation Priority Matrix

Priority Level	Strategy	Success Model
Immediate	Hybrid measurement systems	Radiology AI augmentation
	Standardized validation	Pathology AI validation
Medium-term	Regulatory pathways	FDA-approved frameworks
	Cost-effectiveness demo	Screening AI ROI models
Long-term	Complete spine analysis	Multi-modal integration

3) Immediate actions (1-2 years)

a) Hybrid system development: Research efforts should focus on hybrid system creation that combines traditional geometric methods for reliable measurement with AI enhancement for quality improvement and error detection. This strategy follows successful radiology AI applications where AI supports rather than replaces radiologist skills. Current spine segmentation approaches [62], [63] offer technical foundations for hybrid systems that blend deep learning capabilities with geometric constraints, enabling the enhancement strategy proven successful in other medical AI fields.

b) Validation protocol standardization: Validation protocol standardization represents another key priority. Organizations should create multi-site validation procedures that include real-world imaging variation, following successful pathology AI validation approaches that test systems within actual clinical workflows instead of controlled laboratory settings.

c) Workflow integration research: Implementation research must tackle clinical workflow integration from the start, following the diabetic retinopathy screening model where clinical workflow needs guided technical development rather than being handled as an afterthought.

4) Medium-term objectives (3-5 years)

a) Regulatory pathway development: Work with regulatory agencies to create suitable frameworks for measurement enhancement systems, learning from established FDA routes for diagnostic support AI while addressing the specific needs of measurement improvement applications.

b) Economic value demonstration: Conduct thorough cost-effectiveness studies that show clear ROI for hybrid systems, following successful screening AI models that offer clear value benefits through improved accuracy and workflow efficiency.

c) Clinical evidence generation: Create clinical evidence through pilot implementations that show improved measurement reliability and workflow efficiency, following the evidence development strategies used in successful medical AI fields.

B. Technology Selection Framework Based on Medical AI Success Patterns

Healthcare institutions considering automated system implementation can use our analysis for evidence-based guidance drawn from successful medical AI patterns rather than theoretical performance data.

1) Technology recommendation framework: Based on comprehensive review of 55 studies and comparison with successful medical AI fields, current evidence strongly favors hybrid enhancement approaches over complete automation systems for clinical use.

2) Learning from successful implementation patterns: Successful medical AI fields achieve clinical value through enhancement strategies that improve human capabilities while keeping human oversight and decision control. Scoliosis AI should follow similar approaches, emphasizing measurement quality improvement, error detection, and workflow support instead of pursuing complete measurement automation.

3) Clinical decision support framework: Healthcare institutions should consider enhancement systems for high-volume screening programs needing standardized procedures, quality assurance applications for manual measurement validation, educational environments for training standardization, and research studies requiring measurement standardization and error reduction.

4) Implementation readiness assessment: Healthcare institutions should assess implementation readiness using criteria from successful medical AI deployments. Key evaluation areas include technical infrastructure capable of supporting hybrid workflows, staff training capacity for new enhancement tools, quality assurance procedures adapted for human-AI collaboration, and organizational commitment to workflow adaptation and continuous improvement.

C. Policy and Regulatory Development Based on Medical AI Success

Policy and regulatory development for measurement enhancement system should take lessons from successful medical AI but still adapt to its own technical requirement.

1) Regulatory framework development: Diagnostic AI validation method don't fully work here. New framework should cover training data diversity, algorithm transparency, and performance monitoring. Post-market evaluation must be continuous. Liability keep human in decision loop but also address system error. Benchmarks set minimum measurable accuracy improvement.

2) Funding strategy realignment: Funding should shift to implementation-focused research that show clinical value. Priority include: (i) clinical integration studies from AI success stories, (ii) validation methods for real-world prediction, (iii) hybrid human–AI systems that combine skill and enhancement, and (iv) cross-disciplinary collaboration connecting technical, clinical, and implementation expertise.

3) Technology integration: Integration should follow proven models: multimodal imaging (X-ray, MRI, surface topography) from radiology AI, explainable decision support from pathology AI, and federated learning for multi-site training with privacy protection.

4) Economic model development: Economic structure should follow medical AI demonstration: ROI based on workflow improvement and error reduction, cost-effectiveness beyond setup, and reimbursement aligned with medical AI reimbursement examples.

D. Summary

This analysis, drawn from successful medical AI, points a way forward. By shifting to enhancement, aligning funding, and following proven patterns, scoliosis AI adoption can show improved progress. The recommendation is simple but important: success will come when technology helps the human expert, not when it tries to replace.

VIII. Conclusions

It was found from 55 studies that a paradox appears. Performance has kept improving for 70 years, but clinical usage keeps going down. Semi-automated method performs better than fully automated, with accuracy 1.8° ± 0.4° MAD against 4.2° ± 1.8° MAE. This suggests the opposite of what medical AI normally shows, where more complexity usually performs better. Moreover, moving from lab to clinic cause 5–10% accuracy drop, and clinical use still below 5%. This shows that technical gain does not equal real usage. The main cause is task-technology mismatch. Cobb angles need geometric accuracy, but AI chase full automation instead of enhancement. Other fields already show improved progress with enhancement, but scoliosis AI push too far. This reduces trust, reduces confidence, and reduces adoption. Thus, it can be concluded that success needs shift to enhancement. Hybrid system with mathematical accuracy plus AI support perform better and give clearer value. Institutions should focus on measurement quality, error detection, and workflow support with human control. The message is simple but repeat again: success will not come from replacing the human, it will come from helping the human. In terms of practical steps, implementation needs coordinated action across three phases. The immediate priority is to create hybrid enhancement system and to establish validation procedures that include real-world workflow integration. The medium-term goal is to set regulatory route for measurement enhancement system and to provide cost-effectiveness evidence through pilot implementation. The long-term aim is to integrate multi-modal approach and comprehensive spine analysis capability. Finally, success depends on testing system inside actual clinical workflow rather than only laboratory setting, following the validation pattern already established in successful medical AI. This analysis has several important limitations that should be recognized. The focus on English-language publications may have missed relevant international developments, particularly given the global nature of scoliosis research. Publication bias may favor positive results, potentially overestimating automated system performance compared to clinical reality. The rapid pace of technology change means some recent advances may not yet show their clinical implementation potential, while the 77-year study timeframe may hide recent improvement trends. Additionally, limited availability of real-world clinical validation data restricted the scope of implementation analysis, and the binary view of implementation success versus failure may oversimplify the complex factors affecting clinical adoption. The field must abandon complete automation approaches in favor of augmentation strategies that have proven successful across medical AI domains. Researchers should focus on hybrid system development, clinicians should advocate for workflow-integrated solutions, industry should invest in augmentation rather than replacement technologies, and regulators should develop appropriate frameworks for measurement enhancement systems. Only through coordinated stakeholder efforts that incorporate lessons from successful medical AI implementations while addressing the unique challenges of geometric measurement tasks can the field overcome this translation crisis and deliver meaningful benefits for patient care.

Acknowledgment

The authors gratefully acknowledge all researchers whose work was reviewed in this analysis and thank Penerbit Universiti Malaysia Sabah (UMS Press) for funding.

References

[1] A. B. Ferguson, "The study and treatment of scoliosis," *South Med J*, vol. 23, no. 2, pp. 116-120, 1930.

[2] A. Greenspan, J. W. Pugh, A. Norman, and R. S. Norman, "Scoliotic index: a comparative evaluation of methods for the measurement of scoliosis," *Bulletin of the Hospital for Joint Diseases*, vol. 39, no. 2, pp. 117-125, 1978.

[3] K. M. Diab, J. A. Sevastik, R. Hedlund, and I. A. Suliman, "Accuracy and applicability of measurement of the scoliotic angle at the frontal plane by Cobb's method, by Ferguson's method and by a new method," *European Spine Journal*, vol. 4, pp. 291-295, 1995.

[4] Y. Chen, W. Chen, and W. Chiou, "An alternative method for measuring scoliosis curvature," *Orthopedics-New Jersey*, vol. 30, no. 10, p. 828, 2007.

[5] B. Drerup and E. Hierholzer, "Evaluation of frontal radiographs of scoliotic spines---Part I measurement of position and orientation of vertebrae and assessment of clinical shape parameters," *Journal of Biomechanics*, vol. 25, no. 11, pp. 1357-1362, 1992.

[6] R. T. Morrissy, G. S. Goldsmith, E. C. Hall, D. Kehl, and G. H. Cowie, "Measurement of the Cobb angle on radiographs of patients who have scoliosis. Evaluation of intrinsic error," *JBJS*, vol. 72, no. 3, pp. 320-327, 1990.

[7] C. E. Beekman and V. Hall, "Variability of scoliosis measurement from spinal roentgenograms," *Physical Therapy*, vol. 59, no. 6, pp. 764-765, 1979.

[8] J. E. H. Pruijs, M. A. P. E. Hageman, W. Keessen, R. Van Der Meer, and J. C. Van Wieringen, "Variation in Cobb angle measurements in scoliosis," *Skeletal Radiology*, vol. 23, pp. 517-520, 1994.

[9] D. L. Carman, R. H. Browne, and J. G. Birch, "Measurement of scoliosis and kyphosis radiographs. Intraobserver and interobserver variation," *JBJS*, vol. 72, no. 3, pp. 328-333, 1990.

[10] R. Kundu, A. Chakrabarti, and P. K. Lenka, "Cobb angle measurement of scoliosis with reduced variability," *arXiv preprint arXiv:1211.5355*, 2012.

[11] K. G. Shea, P. M. Stevens, M. Nelson, J. T. Smith, K. S. Masters, and S. Yandow, "A comparison of manual versus computer-assisted radiographic measurement: intraobserver measurement variability for Cobb angles," *Spine*, vol. 23, no. 5, pp. 551-555, 1998.

[12] M. C. Tanure, A. P. Pinheiro, and A. S. Oliveira, "Reliability assessment of Cobb angle measurements using manual and digital methods," *The Spine Journal*, vol. 10, no. 9, pp. 769-774, 2010.

[13] J. Zhang, H. Li, L. Lv, and Y. Zhang, "Computer-aided cobb measurement based on automatic detection of vertebral slopes using deep neural network," *International Journal of Biomedical Imaging*, vol. 2017, no. 1, p. 9083916, 2017.

[14] R. Choi, K. Watanabe, H. Jinguji, N. Fujita, Y. Ogura, S. Demura, T. Kotani, K. Wada, M. Miyazaki, H. Shigematsu, and Y. Aoi, "CNN-based spine and Cobb angle estimator using moire images," *IIEEJ Transactions on Image Electronics and Visual Computing*, vol. 5, no. 2, pp. 135-144, 2017.

[15] M. H. Horng, C. P. Kuok, M. J. Fu, C. J. Lin, and Y. N. Sun, "Cobb angle measurement of spine from X-ray images using convolutional neural network," *Computational and Mathematical Methods in Medicine*, vol. 2019, p. 6357171, 2019.

[16] Y. J. Kim, B. Ganbold, and K. G. Kim, "Web-based spine segmentation using deep learning in computed tomography images," *Healthcare Informatics Research*, vol. 26, no. 1, pp. 61-67, 2020.

[17] B. Ibragimov, B. Likar, F. Pernuš, and T. Vrtovec, "Shape representation for efficient landmark-based segmentation in 3-D," *IEEE Transactions on Medical Imaging*, vol. 33, no. 4, pp. 861-874, 2014.

[18] K. Singer, T. J. Jones, and P. D. Breidahl, "A comparison of radiographic and computer-assisted measurements of thoracic and

thoracolumbar sagittal curvature," *Skeletal Radiology*, vol. 19, pp. 21-26, 1990.

[19] J. Zhang, E. Lou, L. H. Le, D. L. Hill, J. V. Raso, and Y. Wang, "Automatic Cobb measurement of scoliosis based on fuzzy Hough Transform with vertebral shape prior," *Journal of Digital Imaging*, vol. 22, pp. 463-472, 2009.

[20] B. Chen, Q. Xu, L. Wang, S. Leung, J. Chung, and S. Li, "An automated and accurate spine curve analysis system," *IEEE Access*, vol. 7, pp. 124596-124605, 2019.

[21] Z. Wang, Z. Zhang, and I. Voiculescu, "RAR-U-Net: a residual encoder to attention decoder by residual connections framework for spine segmentation under noisy labels," in *2021 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2021, pp. 21-25.

[22] J. H. Han, L. Kóczy, and T. Poston, "Fuzzy hough transform," *Pattern Recognition Letters*, vol. 15, no. 7, pp. 649-658, 1994.

[23] S. Allen, E. Parent, M. Khorasani, D. L. Hill, E. Lou, and J. V. Raso, "Validity and reliability of active shape models for the estimation of Cobb angle in patients with adolescent idiopathic scoliosis," *Journal of Digital Imaging*, vol. 21, pp. 208-218, 2008.

[24] R. Korez, B. Ibragimov, B. Likar, F. Pernuš, and T. Vrtovec, "A framework for automated spine and vertebrae interpolation-based detection and model-based segmentation," *IEEE Transactions on Medical Imaging*, vol. 34, no. 8, pp. 1649-1662, 2015.

[25] B. Glocker, J. Feulner, A. Criminisi, D. R. Haynor, and E. Konukoglu, "Automatic localization and identification of vertebrae in arbitrary field-of-view CT scans," in *Medical Image Computing and Computer-Assisted Intervention--MICCAI 2012: 15th International Conference, Nice, France, October 1-5, 2012, Proceedings, Part III 15*, Springer Berlin Heidelberg, 2012, pp. 590-598.

[26] S. R. Eddy, "Hidden markov models," *Current Opinion in Structural Biology*, vol. 6, no. 3, pp. 361-365, 1996.

[27] B. Schuster-Böckler and A. Bateman, "An introduction to hidden Markov models," *Current Protocols in Bioinformatics*, vol. 18, no. 1, pp. A-3A, 2007.

[28] S. Kónya, T. S. Natarajan, H. Allouch, K. A. Nahleh, O. Y. Dogheim, and H. Boehm, "Convolutional neural network-based automated segmentation and labeling of the lumbar spine X-ray," *Journal of Craniovertebral Junction and Spine*, vol. 12, no. 2, pp. 136-143, 2021.

[29] B. Schlager, M. Krump, J. Boettinger, J. Koenigschulte, S. Doyle, J. Mayr, and R. Wegenkittl, "Deep learning methods for automatic vertebrae identification and morphometric analysis of the spine in MR images," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106230, 2021.

[30] H. Wu, C. Bailey, P. Rasoulinejad, and S. Li, "Automatic landmark estimation for adolescent idiopathic scoliosis assessment using BoostNet," in *Medical Image Computing and Computer Assisted Intervention− MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, Springer International Publishing, 2017, pp. 127-135.

[31] H. Sun, X. Zhen, C. Bailey, P. Rasoulinejad, Y. Yin, and S. Li, "Direct estimation of spinal cobb angles by structured multi-output regression," in *International Conference on Information Processing in Medical Imaging*, Springer International Publishing, Cham, 2017, pp. 529-540.

[32] B. Khanal, L. Dahal, P. Adhikari, and B. Khanal, "Automatic cobb angle detection using vertebra detector and vertebra corners regression," in *International Workshop and Challenge on Computational Methods and Clinical Applications for Spine Imaging*, Springer International Publishing, Cham, 2019, pp. 81-87.

[33] H. S. Kim, S. Ishikawa, Y. Ohtsuka, H. Shimizu, T. Shinomiya, and M. A. Viergever, "Automatic scoliosis detection based on local centroids evaluation on moire topographic images of human backs," *IEEE Transactions on Medical Imaging*, vol. 20, no. 12, pp. 1314-1320, 2001.

[34] P. O. Ajemba, L. Ramirez, N. G. Durdle, D. L. Hill, and V. J. Raso, "A support vectors classifier approach to predicting the risk of progression of adolescent idiopathic scoliosis," *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 2, pp. 276-282, 2005.

[35] S. Rothstock, H. R. Weiss, D. Krueger, and L. Paul, "Clinical classification of scoliosis patients using machine learning and markerless 3D surface trunk data," *Medical & Biological Engineering & Computing*, vol. 58, no. 12, pp. 2953-2962, 2020.

[36] J. Fatima, M. Mohsan, A. Jameel, M. U. Akram, and A. Muzaffar Syed, "Vertebrae localization and spine segmentation on radiographic images for feature-based curvature classification for scoliosis," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 26, p. e7300, 2022.

[37] L. Ramirez, N. G. Durdle, V. J. Raso, and D. L. Hill, "A support vector machines classifier to assess the severity of idiopathic scoliosis from surface topography," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 84-91, 2006.

[38] A. A. Z. Imran, C. Huang, H. Tang, W. Fan, K. Cheung, M. To, Z. Qian, and D. Terzopoulos, "Analysis of scoliosis from spinal x-ray images," *arXiv preprint arXiv:2004.06887*, 2020.

[39] H. Lin, "Identification of spinal deformity classification with total curvature analysis and artificial neural network," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 1, pp. 376-382, 2007.

[40] P. Phan, N. Mezghani, M. L. Nault, C. É. Aubin, S. Parent, J. de Guise, and H. Labelle, "A decision tree can increase accuracy when assessing curve types according to Lenke classification of adolescent idiopathic scoliosis," *Spine*, vol. 35, no. 10, pp. 1054-1059, 2010.

[41] J. Zhang, H. Li, L. Lv, X. Shi, F. Guo, and Y. Zhang, "A Computer-aided Method for Improving the Reliability of Lenke Classification for Scoliosis," *Journal of Healthcare Engineering*, vol. 6, no. 2, pp. 145-158, 2015.

[42] D. Yang, T. T. Y. Lee, K. K. L. Lai, Y. S. Wong, L. N. Wong, J. L. Yang, T. P. Lam, R. M. Castelein, J. C. Y. Cheng, and Y. P. Zheng, "A novel classification method for mild adolescent idiopathic scoliosis using 3D ultrasound imaging," *Medicine in Novel Technology and Devices*, vol. 11, p. 100075, 2021.

[43] D. Liu, L. Zhang, J. Yang, and A. Lin, "Lenke classification of scoliosis based on segmentation network and adaptive shape descriptor," *Applied Sciences*, vol. 13, no. 6, p. 3905, 2023.

[44] L. G. Lenke, "A Decision Tree Can Increase Accuracy When Assessing Curve Types According to Lenke Classification of Adolescent Idiopathic Scoliosis Point of View," *Spine*, vol. 35, no. 10, pp. 1060-1060, 2010.

[45] M. Ogon, K. Giesinger, H. Behensky, C. Wimmer, M. Nogler, C. M. Bach, and M. Krismer, "Interobserver and intraobserver reliability of Lenke's new scoliosis classification system," *Spine*, vol. 27, no. 8, pp. 858-862, 2002.

[46] M. A. Larhmam, S. Mahmoudi, and M. Benjelloun, "Semi-automatic detection of cervical vertebrae in X-ray images using generalized Hough transform," in *2012 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA)*, IEEE, 2012, pp. 396-401.

[47] A. Rasoulian, R. Rohling, and P. Abolmaesumi, "Lumbar spine segmentation using a statistical multi-vertebrae anatomical shape+ pose model," *IEEE Transactions on Medical Imaging*, vol. 32, no. 10, pp. 1890-1900, 2013.

[48] N. Lessmann, B. Van Ginneken, P. A. De Jong, and I. Išgum, "Iterative fully convolutional neural networks for automatic vertebra segmentation and identification," *Medical Image Analysis*, vol. 53, pp. 142-155, 2019.

[49] L. Zhang, L. Shi, J. C. Y. Cheng, W. C. W. Chu, and S. C. H. Yu, "LPAQR-Net: efficient vertebra segmentation from biplanar whole-spine radiographs," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2710-2721, 2021.

[50] L. Zhang, J. Zhang, and S. Gao, "Region-Based Convolutional Neural Network-Based Spine Model Positioning of X-Ray Images," *BioMed Research International*, vol. 2022, no. 1, p. 7512445, 2022.

[51] W. Shi, T. Xu, H. Yang, Y. Xi, Y. Du, J. Li, and J. Li, "Attention gate based dual-pathway network for vertebra segmentation of X-ray spine images," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 8, pp. 3976-3987, 2022.

[52] S. Pang, C. Pang, Z. Su, L. Lin, L. Zhao, Y. Chen, Y. Zhou, H. Lu, and Q. Feng, "DGMSNet: Spine segmentation for MR image by a detection-guided mixed-supervised segmentation network," *Medical Image Analysis*, vol. 75, p. 102261, 2022.

[53] M. U. Saeed, N. Dikaios, A. Dastgir, G. Ali, M. Hamid, and F. Hajjej, "An automated deep learning approach for spine segmentation and

vertebrae recognition using computed tomography images," *Diagnostics*, vol. 13, no. 16, p. 2658, 2023.

[54] R. Da Mutten, O. Zanier, S. Theiler, S. J. Ryu, L. Regli, C. Serra, and V. E. Staartjes, "Whole Spine Segmentation Using Object Detection and Semantic Segmentation," *Neurospine*, vol. 21, no. 1, p. 57, 2024.

[55] N. Chockalingam, P. H. Dangerfield, G. Giakas, T. Cochrane, and J. C. Dorgan, "Computer-assisted Cobb measurement of scoliosis," *European Spine Journal*, vol. 11, pp. 353-357, 2002.

[56] H. Li, C. Qian, W. Yan, D. Fu, Y. Zheng, Z. Zhang, J. Meng, and D. Wang, "Use of Artificial Intelligence in Cobb Angle Measurement for Scoliosis: Retrospective Reliability and Accuracy Study of a Mobile App," Journal of Medical Internet Research, vol. 26, p. e50631, 2024.

[57] Y. Maeda, T. Nagura, M. Nakamura, et al., "Automatic measurement of the Cobb angle for adolescent idiopathic scoliosis using convolutional neural network," Scientific Reports, vol. 13, p. 14576, 2023.

[58] V. K. Bürger, J. Amann, C. K. T. Bui, J. Fehr, and V. I. Madai, "The unmet promise of trustworthy AI in healthcare: why we fail at clinical translation," Frontiers in Digital Health, vol. 6, p. 1279629, 2024.

[59] E. Ahmed, R. Ward, S. Delves, B. Spooner, J. Isherwood, A. Dhami, S. Poyser, and J. Ghosh, "A Systematic Review of the Barriers to the Implementation of Artificial Intelligence in Healthcare," Cureus, vol. 15, no. 10, p. e46454, 2023.

[60] A. Suri, S. Tang, D. Kargilis, E. Taratuta, B. J. Kneeland, G. Choi, A. Agarwal, N. Anabaraonye, W. Xu, J. B. Parente, A. Terry, A. Kalluri, K. Song, and C. S. Rajapakse, "Conquering the Cobb Angle: A Deep Learning Algorithm for Automated, Hardware-Invariant Measurement of Cobb Angle on Radiographs in Patients with Scoliosis," Radiology: Artificial Intelligence, vol. 5, no. 4, p. e220158, 2023.

[61] W. Caesarendra, W. Rahmaniar, J. Mathew, and A. Thien, "Automated Cobb Angle Measurement for Adolescent Idiopathic Scoliosis Using Convolutional Neural Network," Diagnostics, vol. 12, no. 2, p. 396, 2022.

[62] M. U. Saeed, N. Dikaios, A. Dastgir, G. Ali, M. Hamid, and F. Hajjej, "An automated deep learning approach for spine segmentation and vertebrae recognition using computed tomography images," Diagnostics, vol. 13, no. 16, p. 2658, 2023.

[63] H. Lu, M. Li, K. Yu, Y. Zhang, and L. Yu, "Lumbar spine segmentation method based on deep learning," Journal of Applied Clinical Medical Physics, vol. 24, p. e13996, 2023.

Adaptive Trust-Based Fault Tolerance for Multi-Drone Systems: Theory and Application in Agriculture

Atef GHARBI[1], Faheed A.F. Alrslani[2]
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]

***Abstract*—This paper presents RobotTrust, an adaptive trust framework for fault-tolerant coordination in multi-drone systems for precision agriculture. The study aims to improve mission reliability under sensor/actuator faults and uncertain interactions by combining a structured fault taxonomy (behavioral, actuator, sensor) with team-based recovery and an adaptive trust model that integrates direct experience with filtered indirect recommendations. We formalize trust computation (direct, recommended, and global trust) and introduce safeguards such as a minimum-trust threshold and weighted fusion to curb misinformation propagation. The framework is evaluated in simulation using the AgriFleet drone team and is compared against the TReconf baseline across three metrics: (i) time-step efficiency for task completion, (ii) RMSD between predicted and true trustworthiness, and (iii) interaction quality (preference for reliable peers). Results show 20–40% faster task completion, lower RMSD (more accurate trust estimation), and selective interaction patterns that prioritize dependable agents while limiting exposure to unreliable ones. These findings indicate that RobotTrust enhances responsiveness and robustness in decentralized, fault-prone environments typical of agricultural deployments. The work contributes a practical, generalizable approach to trust-aware coordination in multi-robot systems and outlines directions for context-aware weighting, explainable trust signals, heterogeneous teams, adversarial robustness, and large-scale field trials.**

Keywords*—*Adaptive trust model; trust-aware robotics; multi-drone coordination; fault-tolerant systems; precision agriculture applications

I. Introduction

Multi-Robot Systems (MRS) are new approaches in automation and enable collaborative tasks in complex and dynamic environments, such as precision agriculture [1-3], search and rescue missions [4-6], and industrial logistics [7-8]. These systems provide significant scalability, redundancy and adaptation advantages using distributed intelligence, qualities that are essential to real-world applications. However, its decentralized nature poses crucial challenges in fault management and trust-based coordination, where unrecognized failures or poor communication can lead to systemic inefficiencies or total failures of missions [9-11]. The latest progress in MRS focuses on improving fault tolerance and cooperative decision-making. Traditional fault detection methods, such as rule-based diagnostics [12] and anomaly detection based on machine learning [13], are widely accepted. These approaches are effective in controlled environments, but these solutions often lack the sensitivity required for dynamic environments where robot interactions are intrinsically uncertain. At the same time, the trust model in MRS has evolved significantly, and frameworks such as TReconf [14] provide mechanisms for reliability assessment. However, these models demonstrated limitations in dealing with real-time adaptation and reducing errors caused by misinformation, especially in systems affected by sensor noise or partial observation [15]. The recent work of Li et al. [16] has made progress in the integration of trust and fault recovery, but there is still a lack of a comprehensive framework for combining dynamic fault classification, team-based recovery and adaptive trust calculation.

While existing models like TReconf offer basic trust-based coordination, they fall short in handling real-time fault adaptation and mitigating misinformation spread, especially under sensor noise and partial observability. This gap underscores the need for a more robust framework that integrates dynamic fault classification with adaptive trust evaluation—motivating the development of RobotTrust. This paper presents an integrated framework for fault-resistance MRS, addressing these challenges. Our approach begins with a systematic fault classification system that classifies faults into behavioral, actuator and sensor errors, enabling targeted diagnosis and recovery. Based on this, we have developed a decentralized fault handling platform, which uses dynamic task assignment, shared knowledge and self-diagnosis mechanisms to maintain the continuity of the mission. The core innovation of the framework is the RobotTrust model, an adaptive trust evaluation system that quantifies robot reliability through direct interactions, transitional recommendations and global trust aggregation, significantly improving resilience to sensor failures and coordination failures. We validate our framework by extensive testing with AgriFleet, a multi-drone platform designed for precision agricultural applications. Our comparative analysis compares RobotTrust with the established TReconf model [14] in scenarios that simulate the probability of defects and produces three key results. First, our approach has shown that there are approximately 20% improvements in the efficiency of the time-step for the

completion of tasks. Secondly, it reaches greater accuracy of trust, as shown by lower values of root average square deviation than TReconf. Thirdly, the system effectively reduces unnecessary interactions with unreliable agents and optimizes the overall performance of the system.

Our work has made four important contributions in this field. We introduced a structured fault taxonomy for MRS, supported by real-world case studies of the AgriFleet platform. We have developed a new team-based fault recovery mechanism to ensure robust operation in dynamic environments. The RobotTrust model represents a significant advance over existing trust computation frameworks. Finally, our extensive empirical validation shows that the system is more resistant to errors under fault-prone operation conditions.

This paper is organized as follows. Section II provides a detailed examination of fault classification and handling strategies. Section III introduces the computational framework underlying RobotTrust. Section IV presents our comparative simulation results and performance analysis. By integrating fault resilience with trust-aware coordination, this work advances the state of MRS robustness, offering practical solutions that address real-world deployment challenges.

II. Collaborative Fault Management Framework for Multi-Robot Systems

Exploration of multi-robot systems (MRSs) requires proactive fault management, especially by early identification and classification of possible system failures. Establishing clear fault categories provides the basis for developing a structured response strategy. In this context, we present a collaborative fault management framework that integrates team-based coordination to detect, isolate and mitigate faults between multiple robots. This strategy emphasizes the system's collective capability and promotes resilience and maintenance of functionality even in the case of failure of individual components.

A. Fault Classification

To enable systematic fault detection and processing, we classify faults that affect intelligent robot performance in MRS environments into three main areas: behavioral, actuator and sensor faults.

1) Behavioral deficiencies: Behavioral faults refer to the decision-making and planning processes of robots:

a) Action deficiencies: occur when a robot does not perform a commanded action correctly or completely.

b) Plan fault: a failure of a planned action plan caused by a robot not achieving the intended goal.

c) Unexpected conditions: When robots encounter unpredicted environmental conditions and tasks outside their operational design, triggers are activated.

2) Actuator errors: These errors affect the robot's ability to interact physically with the environment:

a) Blocked actuator errors: actuators cannot initiate the necessary movements or operations despite receiving a command.

b) Blocking fault: Even in the absence of a control signal, the actuator remains inactive or active.

3) Sensor errors: These involve errors in the acquisition of sensory data, affecting situational consciousness:

a) Sensor bias errors: sensors provide constant dispersion or misreading, which affects perception and decision-making.

b) Sensor freezing fault: The sensor becomes unresponsive, ceasing to update or transmit data.

B. Fault Handling Using a Team Approach

In collaborative MRS, effective fault management and dynamic task coordination are essential, especially when individual agents encounter operational failures during collective missions. Consider the scenario in which robot R1 suffers a failure (F1) while performing a collective task. The fault management process involves several key participants: a defective robot (R1), other participating robots (such as R2, R3), and a central coordinate entity called a fault manager. Two main system components support this architecture:

- Shared knowledge: A repository containing the team's current understanding of tasks objectives and allocations.
- The Fault Registry is a structured record of known faults, related robots, and current resolution status.

Communication between system components is promoted by structured messages, including report Fault, query Capabilities, updateTaskPlan, performTask, diagnoseFault, reportStatus, and updateFaultRegistry.

III. Multi-Robot Systems Trust Modeling

This section presents a new trust assessment framework, RobotTrust, and compares it to the existing TReconf trust model described in study [14]. The robotic trust methodology is described in detail, focusing on its application to fault detection and trust-based decision making in multi-robot systems (MRSs). The RobotTrust model is an effective mechanism for assessing the trustworthiness of robots in a multi-agent environment. It facilitates the monitoring and detection of faulty robots by analyzing historical interaction data. The model computes multiple trust indicators, including direct trust, recommended trust, and global trust values, and provides a multilevel understanding of the reliability between robots. Trust (i) represents the global trust value assigned to robot i, and Trust (i, j) represents the local trust value assigned to robot j by robot i in direct or indirect interactions. The trust model distinguishes between robots that have previously interacted and those that have not, structuring trust computation accordingly.

A. Types of Trust Metrics

1) Direct trust value: The direct trust value quantifies the relationship between two robots engaged in observable interactions. Using the difference between the number of satisfactory and unsatisfactory interactions as shown in Eq. (1).

$$S_{a,b} = \text{sat}(a,b) - \text{unsat}(a,b) \quad (1)$$

This defines the direct trust between robots (a) and (b), as defined in Eq. (2).

$$trust(a,b) = f(S_{a,b}) \quad (2)$$

and f represents a transformation function that normalizes the satisfaction score.

2) Recommended trust value: If there is no direct interaction between two robots (e.g. (a) and (d)), the recommended trust value is derived using transition relationships defined in Eq. (3).

$$Trust(a,d) = Trust(a,k) \times Trust(k,d) \quad (3)$$

$$\text{where } k \in \text{DirectPeers}(a) \cap \text{DirectPeers}(d)$$

This method uses mutual neighbors to estimate the trust between agents indirectly connected.

3) Global trust value: Global trust value represents the trustworthiness of a robot in the network, aggregating direct and recommended trust metrics of all other robots, as shown in Eq. (4).

$$T_i^{k+1} = \sum_{j=1}^{n} Trust(j,i) \cdot T_j^{k+1} \quad (4)$$

where:

- T_i^{k+1}: global trust of a robot i at the k+1 iteration,
- Trust(j,i): local trust of a robot from j to i,
- n: total number of robots in the network.

Firstly, the global trust values of all robots are distributed uniformly, i.e. for N robot networks. Algorithm 1 calculates the adaptive trust between robots in two main phases: the direct trust calculation of immediate peers and the indirect trust propagation of non-direct peers. Firstly, all trust values are initialized as neutral values (0.5), which represent uncertainty regarding unknown peers. For each direct peer (the robot j with which robot i has directly interacted), the algorithm calculates a trust score based on the historical interaction. It counts a satisfactory (pos) and an unsatisfactory (neg) interaction, and then combines these interactions with a parameter α (set to 0.7). Trust value mixes the satisfaction ratio (pos/total interaction) and the penalty factor (1/(1+neg)), which is degraded with negative experiences. The result is limited between 0 and 1 to ensure a valid trust value. If robot R_1 (trust = 0.9) shares accurate data, its trust score increases; if robot R_2 (trust = 0.6) reports false data, its score decays.

In the case of the indirect peers (robot j with which robot i has not directly interacted), the algorithm calculates trust through intermediary robots connecting i and j. It only considers intermediaries that meet the minimum trust threshold (min_trust = 0.1) to prevent unreliable spread. The indirect trust is calculated as a weighted average of the path of trust, and the contribution of each path depends on the relationship between trust (i,k) and trust (k,j). Parameter β (0.3) controls the amount of indirect trust that updates the existing trust value. If there is no valid path, trust remains at initial neutral value. The algorithm generates a trust value of 0 (complete distrust) to 1 (complete trust) and combines direct experience with carefully filtered indirect information to form a complete assessment of each peer's trust.

Algorithm 1: ComputeAdaptiveTrust

```
Output:
Trust(i,j): trust value assigned by robot i to each robot j
Parameters:
 α ← 0.7   // weight for direct experience
 β ← 0.3   // weight for indirect trust propagation
min_trust ← 0.1   // minimum trust threshold for propagation
neutral_trust ← 0.5   // default neutral trust value

Initialize Trust(i,j) ← neutral_trust for all robot j ∈ AllRobots
// First calculate direct trust
for each robot j ∈ DirectPeers(i) do
    pos ← countSatisfactory(i, j)
    neg ← countUnsatisfactory(i, j)
    total ← pos + neg
     if total > 0 then
         satisfaction_ratio ← pos / total
         penalty_factor ← 1 / (1 + neg)
         Trust(i,j) ← α * satisfaction_ratio + (1 - α ) *
penalty_factor
         Trust(i,j) ← max(0, min(1, Trust(i,j)))
    end if
end for

// Calculate indirect trust for non-direct peers
for each robot j ∈ IndirectPeers(i) \ DirectPeers(i) do
    candidates ← findBridgeRobots(i, j)
    weighted_trust_sum ← 0
    total_weights ← 0
    for each robot k ∈ candidates do
        // Only with sufficient trust
        if Trust(i,k) ⩾ min_trust and Trust(k,j) ⩾ min_trust
            weight ← Trust(i,k) * Trust(k,j)
           weighted_trust_sum ← weighted_trust_sum +
weight
            total_weights ← total_weights + weight
        end if
    end for
    if total_weights > 0 then
        indirect_trust ← weighted_trust_sum / total_weights
        // Combine with existing trust if any
        Trust(i,j) ← β * indirect_trust + (1 - β ) * Trust(i,j)
    end if
  end for
```

IV. Simulation Scenario and Comparative Trust Evaluation

A. AgriFleet: Trust-Based Fault-Tolerant Drone Coordination

The AgriFleet system consists of a team of specially equipped autonomous air drones (D1, D2, D3, and D4), each equipped with specific capabilities required for the precise

surveillance of agriculture and field data collection. D1 performs high-resolution aerial imaging, D2 achieves soil moisture detection, D3 makes multispectral crop health analysis, and D4 performs environmental mapping and terrain classification. Each drone operates independently, but its operational interdependence is essential for the success of complex agricultural tasks. During the execution of missions, drones constantly communicate, and delegate tasks based on environmental data in real time. For example, D1 can detect crop stress signs by image and transmit this information to D3, which then performs detailed spectral analysis of the affected area. Similarly, D3 can identify areas of suspected soil dilution and prompt D2 to carry out targeted moisture assessment. This type of cooperation enables better coverage, avoids redundancy and increases the overall efficiency of the air monitoring process. To ensure consistency of coordination and reliable task execution, the system incorporates dynamic trust models, based on the accuracy and utility of shared data, where the trust value of drones evolves. The level of trust is quantified on a continuous scale of [0.0, 1.0] and 0.5 represents a neutral baseline. Accurate assessments and successful completion of tasks lead to higher confidence, whereas false reports or failures lead to a decrease in trustworthiness. Despite structured cooperation, drones may suffer from sensor failures, each with unique probability of occurrence. These deficiencies are unknown to drones themselves but affect how their output is trusted by peers. The probability of sensor failure and the corresponding initial trust score are presented in Table I.

TABLE I. DRONE SENSOR RELIABILITY AND TRUSTWORTHINESS

Drone	Function	Sensor malfunction	Prob Trust
D1	High-resolution Imaging	0.1	0.9
D2	Soil Moisture Detection	0.4	0.6
D3	Spectral Crop Health Assessment	0	1
D4	Terrain Mapping and Profiling	0.3	0.7

We evaluated the efficiency of RobotTrust models compared to previously developed TReconf model [14]. The objective is to determine how each trust framework adapts to an environment characterized by interdependent behaviour and probabilistic misinformation due to sensor errors. In 85 interaction rounds, drones act as both trustors and trustees, engaging in communication and task delegation, while considering the probability of sensor-induced false information. The formation of trust is influenced by the interaction between trust scores, feedback on success and interaction results in this time window.

B. Comparative Evaluation of RobotTrust and TReconf Models

The performance of the RobotTrust and TReconf models is evaluated based on three basic parameters:

- Time-step analysis: measures the system's response and task completion speed under different trust conditions.
- Root Mean Square Deviation (RMSD): Evaluate the coherence and convergence of trust assessments throughout the simulation timeline.
- Interaction analysis: investigating the quality and frequency of interactions between robots, especially under uncertainties and partial trust conditions.

This comparative analysis aims to determine which models better capture trust dynamics and improve system reliability in collaborative, fault-prone and multi-robot environments.

1) Time step analysis for trust-based aerial coordination: Fig. 1 shows a clear performance advantage over the TReconfig baseline in various simulation iterations. As shown by time-step analysis, RobotTrust consistently requires fewer time-steps to complete coordination tasks, indicating better system reaction and efficient task execution under the proposed trust-based framework. The data reveal several important trends. First, the performance gap between RobotTrust and TReconfig is most prominent during mid-range iterations (40-60 simulation cycles), and the trust model seems to maximize direct experience evaluation and indirect trust propagation. This suggests that adaptive weighting mechanisms ($\alpha = 0.7$ for direct trust, $\beta = 0.3$ for indirect trust) filter unreliable information effectively under typical operational conditions. Secondly, although the performance of the two systems has increased considerably in higher iteration (approximately 80), RobotTrust maintains an important advantage in the testing range. The relative stability of the RobotTrust time step requirement shows that its fault-tolerant design successfully mitigates coordination failures that increasingly impact baseline systems as the complexity of operations increases. The reduced time step of RobotTrust can bring tangible operational benefits to agricultural applications. More short-term work is required, which means faster response to field conditions, more efficient allocation of resources, and ultimately higher completion rates of critical operations such as crop monitoring or precision spraying. These results validate two key characteristics of RobotTrust design: (1) Minimum Trust Limit (min_trust = 0.1) effectively prevents the spread of errors by unreliable intermediaries; and (2) a mixed trust measurement combining satisfaction rates and penalty factors maintains accuracy despite changes in environmental conditions.

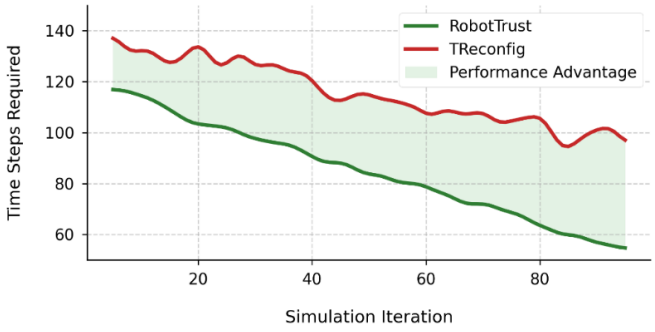


Fig. 1. Comparative analysis of time step efficiency: robottrust vs. treconf in the presence of probabilistic sensor malfunctions.

Performance advantages are particularly present in the most common operational scenario, while modest convergence in

extreme iterations suggests areas for potential improvements in the handling of edge cases. Experimental evidence supports the adoption of trust-based coordination in the agricultural drone fleet, where unpredictable field conditions and the reliability of equipment require a robust and responsive decision-making process.

2) RMSD Evaluation for trustworthiness estimation: The root mean square deviation (RMSD) measurement is used in this study to quantitatively measure the accuracy with which a trust model estimates the true trustworthiness of individual drones in the AgriFleet system. RMSD provides statistical assessments of differences between predictability values generated by trust models and real trust levels derived from established drone attributes such as sensor reliability, mission coherence, and functional performance (as described below). Lower RMSD values indicate a higher degree of alignment between predicted and actual trustworthiness, indicating more accurate and reliable trust models. RMSD is particularly useful in identifying cases of overestimation or underestimation, which may have adverse effects on the coordination of multi-drones, the allocation of resources and the robustness of operations of agricultural missions. For each drone in AgriFleet, the actual value of trust is set as the ground truth reference based on predefined operational parameters. The predicted trust score is derived from the results of the RobotTrust and TReconf models after simulations. The RMSD is computed as shown in Eq. (5).

$$RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(T_{\text{pred},i} - T_{\text{actual},i})^2} \quad (5)$$

Where:

$T_{pred,i}$: Predicted trust value of drone i at the end of simulation,

$T_{actual,i}$: Actual trust value of drone i based on initial drone specifications,

n: Total number of drones (in this case, 4).

Fig. 2 shows a comparison analysis of the coordination performance of the RobotTrust system with TReconf baseline, measured with the root mean square deviation (RMSD) value between different combinations of drones (T1,2 to T3,4). RMSD measurement serves as a quantitative indicator of coordination accuracy, and the lowest value represents more precise and stable collaboration behaviors between drone pairs. The results show that RobotTrust consistently achieves better coordination performance than TReconf in all tested drone pairings. The RMSD values show particularly significant improvements in certain pair combinations (especially T1,3 and T2,4), suggesting that a trust-based approach can effectively address challenging coordination scenarios in which conventional methods fail. This increased performance is based on RobotTrust's adaptive trust mechanism, which dynamically adjusts interaction weights based on real-time reliability assessments. The variation in RMSD reductions in different drone pairs highlights the effectiveness of trust models based on context. Some pairings have seen dramatic improvements, while others have seen more modest but still significant gains. This pattern suggests that, although the trust framework provides universal benefits, its influence is more significant in situations that require complex interdependent procedures or those that are vulnerable to communication failures. These findings confirm the robustness of the Decentralized Coordination Approach of RobotTrust, especially for agricultural applications, where environmental factors and equipment reliability may vary significantly. The continuous reduction in RMSD in all tested pairs confirms that the system maintains its coordination accuracy advantage, regardless of which specific drones are involved, and proves the reliability of the trust framework. The results highlight the practical value of incorporating adaptive trust indicators into multi-drone systems to improve operation.

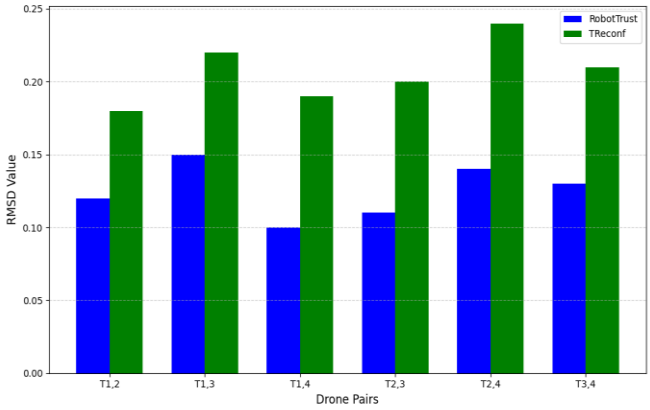


Fig. 2. Root Mean Square Deviation (RMSD) of trust estimation.

3) Interaction analysis: The effectiveness of trust development in multi-drone agricultural systems such as AgriFleet depends primarily on the frequency and quality of interaction between the drones. In this context, interaction analysis evaluates the effectiveness of each trust model (RobotTrust and TReconf) in identifying and prioritizing reliable agents to unreliable agents in field operations.

An optimal trust framework must enable the system to:

- Maximize the value of interactions with reliable drones that are continuously reliable;
- Minimize the dependence on drones that are vulnerable to defects or misinformation;
- Ensure mission progress through reliable communication links for accurate and timely decision-making.

Reliable drones are identified through consistent, accurate and mission-enhancing contributions in various interactions. The trust model that gives these drones a clear priority helps reduce flight redundancy, avoid conflicting data collection, and simplify the overall performance of missions. In the AgriFleet system, the D1 and D3 drones showed higher interaction frequencies than D2 and D4. This distribution is closely linked to the lower probability of sensor malfunction, making it a more reliable data source for collaborative air missions.

Fig. 3 shows the comparison of the thermal maps of the frequencies of interaction between drones under two models: TReconf and RobotTrust. Each subplot shows a 4x4 matrix that represents the number of times each drone interacts with each other, and the rows and columns represent drones D1 to D4. Because drones do not interact with each other, the diagonal entry is zero.

In the TReconf model, interaction frequencies appear to be widely distributed and have relatively high and consistent values for all drone pairs. This indicates a more uniform or dispersed communication model, indicating that TReconf does not differentiate between reliable and unreliable agents in its trust decisions. For example, D1 interacts frequently with other users, including D4, and is unsuitable if D4 is untrustworthy. On the other hand, the RobotTrust model shows a more selective pattern of interaction. The strongest interactions are concentrated in specific pairs (D1 and D3) and show much higher values (e.g. 18 interactions). Interactions with D2 and D4 are less frequent across the board, indicating that RobotTrust is actively avoiding collaborations with drones that might exhibit unreliable behavior. This behavior reflects RobotTrust's intention to isolate unreliable agents and prioritize reliable communication channels. Overall, Fig. 3 effectively shows how trust-based decision-making strategies can shape multi-agent robotic communication patterns.

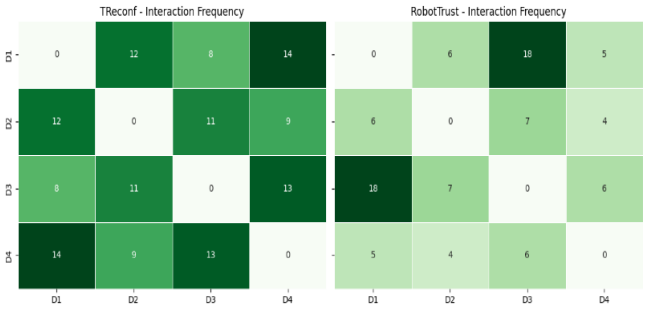


Fig. 3. Heatmap of interaction frequencies among drones under sensor faults, comparing TReconf and RobotTrust models.

C. Discussion

This study shows that RobotTrust improves multi-drone coordination by reducing time-steps to task completion, aligning estimated trust more closely with ground truth (lower RMSD), and prioritizing interactions with reliable peers. These gains appear to stem from two design choices: (i) weighting direct experience more than indirect recommendations and (ii) filtering low-quality recommendations via a minimum-trust threshold. Practically, this combination limits the propagation of misinformation under sensor noise and helps maintain mission progress in fault-prone conditions typical of precision agriculture.

There are, however, several alternative paths we could have taken. First, rather than a hand-crafted trust update, a learned model (e.g., graph neural networks or Bayesian filters) could infer trust dynamics from data, potentially capturing richer inter-robot dependencies at the cost of interpretability and data requirements. Second, instead of rule-based task reassignment, a planning layer based on reinforcement learning or POMDPs could map trust states to actions; this may improve adaptability but raises concerns about sample efficiency and safety guarantees. Third, our probabilistic fault model could be complemented by domain-randomized simulators or hardware-in-the-loop experiments to narrow the sim-to-real gap. Finally, comparisons against consensus-based reputation or Byzantine-resilient aggregation would help position RobotTrust among broader trust frameworks.

We also acknowledge limitations and threats to validity. Communication latency, wind disturbances, GPS/IMU drift, and battery degradation were simplified, and fixed α/β weights may underperform under rapidly changing conditions. Our small-team evaluations (four drones) limit claims about scalability and heterogeneity (air/ground/IoT), and we considered stochastic faults rather than coordinated adversaries.

V. Conclusion

The paper introduced an adaptive trust model to address the fault tolerance coordination of agricultural multidrone systems. The proposed approach enables responsive decision-making even when the individual agent fails or provides unreliable data by dynamically balancing direct experience and indirect trust recommendations through a weighted mechanism. By implementing minimum trust threshold, the system eliminates non-trustworthy intermediaries and significantly reduces the spread of errors across the network. Across simulation studies with the AgriFleet team (D1–D4), RobotTrust reduced coordination time-steps by ~20–40% relative to TReconf, lowered RMSD in trust estimation (indicating closer alignment to true reliability), and prioritized interactions with reliable agents while limiting exposure to unreliable peers. Design elements such as a minimum trust threshold and weighted fusion (α for direct, β for indirect) were pivotal in filtering misinformation and sustaining mission progress under sensor faults. Collectively, these results demonstrate that adaptive, trust-aware coordination can improve responsiveness, reliability, and operational efficiency in fault-prone field conditions typical of agricultural deployments. By accelerating task completion, improving trust accuracy, and curbing counterproductive interactions, RobotTrust supports safer, more efficient flight operations (e.g., crop monitoring, targeted spraying), and offers a generalizable mechanism for decentralized fault management in other mission-critical, multi-robot settings.

Future work will focus on making RobotTrust more adaptive, explainable, and field-ready: first, we will replace fixed trust weights with context-aware weighting that adapts to mission urgency, environmental uncertainty, and sensor health; second, we will add online fault-severity estimation and dynamic task reallocation to shorten recovery latency during actuator/sensor anomalies; third, we will introduce explainable trust signals so operators can audit why trust increased or decreased for a given agent; fourth, we will validate on heterogeneous teams (air/ground/IoT) and new domains beyond agriculture (e.g., disaster response, autonomous logistics) to assess transferability; fifth, we will harden the framework against adversarial misinformation and Byzantine behavior using reputation damping and consensus safeguards; and finally, we will run scalable field trials with larger fleets to

measure compute/communication overheads, energy usage, and mission-level KPIs under realistic fault conditions.

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-2025-2441-02".

REFERENCES

[1] Del Cerro, J., Cruz Ulloa, C., Barrientos, A., & de León Rivas, J. (2021). Unmanned aerial vehicles in agriculture: A survey. Agronomy, 11(2), 203.

[2] Velusamy, P., Rajendran, S., Mahendran, R. K., Naseer, S., Shafiq, M., & Choi, J. G. (2021). Unmanned Aerial Vehicles (UAV) in precision agriculture: Applications and challenges. Energies, 15(1), 217.

[3] Alotaibi, F., Al-Dhaqm, A., & Al-Otaibi, Y. D. (2023). A conceptual digital forensic investigation model applicable to the drone forensics field. Engineering, Technology & Applied Science Research, 13(5), 11608-11615.

[4] Murphy, R. R. (2004). Human-robot interaction in rescue robotics. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 34(2), 138-153.

[5] Chan, T. H., Halim, J. K. D., Tan, K. W., Tang, E., Ang, W. J., Tan, J. Y., ... & Foong, S. (2023, June). A Robotic System of Systems for Human-Robot Collaboration in Search and Rescue Operations. In 2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) (pp. 878-885). IEEE.

[6] Pan, D., Zhao, D., Pu, Y., Wang, L., & Zhang, Y. (2024). Use of cross‐training in human‐robot collaborative rescue. Human Factors and Ergonomics in Manufacturing & Service Industries, 34(3), 261-276.

[7] Dzedzickis, A., Subačiūtė-Žemaitienė, J., Šutinys, E., Samukaitė-Bubnienė, U., & Bučinskas, V. (2021). Advanced applications of industrial robotics: New trends and possibilities. Applied Sciences, 12(1), 135.

[8] Ebel, H. (2021). Distributed control and organization of communicating mobile robots: design, simulation, and experimentation.

[9] Nandanwar, A., Tripathi, V. K., & Behera, L. (2021, July). Fault-tolerant control for multi-robotics system using variable gain super twisting sliding mode control in cyber-physical framework. In 2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) (pp. 1147-1152). IEEE.

[10] Govoni, L., & Cristofaro, A. (2023, July). A fault-tolerant task allocation framework for overactuated multi-robot systems. In 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT) (pp. 287-292). IEEE.

[11] Ding, S. X. (2021). Advanced methods for fault diagnosis and fault-tolerant control (Vol. 184). Berlin/Heidelberg, Germany: Springer.

[12] Ai, X., Liu, Y., Shan, L., Xie, C., & Zhou, H. (2024). A concurrent fault diagnosis method for electric isolation valves in nuclear power plants based on rule-based reasoning and data-driven methods. Progress in Nuclear Energy, 171, 105190.

[13] Li, G., & Jung, J. J. (2023). Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges. Information Fusion, 91, 93-102.

[14] Gharbi, A., & Altowaijri, S. M. (2023). A multi-robot-based architecture and a trust model for intelligent fault management and control systems. Electronics, 12(17), 3679.

[15] Billaud, O., Porcher, E., & Maclouf, E. (2025). Experience and trust to handle uncertainty of biodiversity responses to innovative agricultural practices: Lessons from citizen science on farms. People and Nature.

[16] Guo, Y., Pang, Y., Lyons, J., Lewis, M., Sycara, K., & Liu, R. (2024). Trust-Aware Reflective Control for Fault-Resilient Dynamic Task Response in Human–Swarm Cooperation. AI, 5(1), 446-464.

Method for Person Re-Identification with 2D-to-3D Image (Image-to-Video) Conversion

Kohei Arai
Science and Engineering Faculty, Saga University, Saga City, Japan

***Abstract*—A method for person re-identification using image-to-video conversion tools is proposed. The proposed method involves matching two images taken from different viewpoints: a reference image captured in advance and a current image captured in real-time to identify the person in concern in the current image, whose image is matched to the reference image. The 2D current image is converted into a 3D representation, from which synthetic images are generated at multiple viewpoints. By comparing the generated images and the reference image, person re-identification can be done. Experiments have demonstrated that the proposed method significantly improves identification accuracy. By accounting for changes in appearance due to different viewpoints and utilizing advanced image conversion, one of the main challenges in person re-identification is addressed. This approach offers a promising solution for applications requiring high accuracy in identifying individuals across varying perspectives.**

***Keywords*—Person re-identification; identification performance; 2D-to-3D image conversion method; TRIPO; CSM; KLING**

I. Introduction

Person re-identification involves recognizing an individual across different images or video frames, typically within an anonymized dataset. This task is crucial in the fields of privacy protection and data security. However, existing methods often struggle with low accuracy because a person's appearance can vary significantly when viewed from different angles, and there may be many individuals with similar appearances in the dataset.

Challenges in person re-identification affecting recognition rates are as follows:

Person re-identification (Re-ID) is a crucial task in computer vision and surveillance, aiming to recognize and match individuals across different images or video frames, often captured in varying conditions and from non-overlapping camera views. Despite extensive research, achieving high recognition rates in person Re-ID remains challenging due to several factors:

1) Appearance variations:

a) Viewpoint differences: The same person can appear vastly different when viewed from various angles. Changes in posture, orientation, and perspective can significantly alter visual features.

b) Lighting conditions: Variations in lighting can affect the perceived color and texture of clothing, leading to inconsistent appearance across images.

c) Attire and accessories: Changes in clothing, addition or removal of items like jackets or bags, and seasonal attire shifts (e.g., winter coats versus summer wear) can make the same individual look different.

d) Personal belongings: The presence or absence of belongings such as umbrellas, luggage, or hats can alter the silhouette and features used for identification.

2) Constraints of the shooting environment

a) Low-resolution images: Surveillance cameras often produce low-resolution images, making it challenging to capture fine-grained features necessary for accurate identification.

b) Camera calibration differences: Variations in camera settings can lead to color tone mismatches and distortions, complicating cross-camera recognition.

c) Occlusions: Partial obstructions from objects, other people, or environmental factors can result in only parts of a person being visible, affecting feature extraction.

3) Dataset limitations

a) Insufficient real-world coverage: Collecting data that encompasses the vast diversity of real-world situations is difficult, leading to models that may not generalize well.

b) Privacy concerns: Legal and ethical considerations limit the ability to build large-scale datasets with varied identities, impacting the amount of available training data.

c) Lack of temporal and spatial diversity: Gathering image pairs of the same person at different times and locations is challenging, reducing the effectiveness of models in varying conditions.

4) Algorithmic challenges

a) Feature robustness: Developing algorithms that can extract features resilient to changes in posture, viewpoint, and occlusions is complex.

b) Overreliance on variable features: Many models depend heavily on easily changeable features like clothing color and hairstyle, which can lead to misidentification.

c) Long-term appearance changes: Coping with significant appearance changes over time, such as weight fluctuation or ageing, adds another layer of difficulty.

To overcome the problems, a method for person re-identification using image-to-video conversion tools is proposed in this paper. The proposed method involves matching two images taken from different viewpoints: a reference image captured in advance and a current image captured in real-time. To identify the person in concern in the

current image whose image is matched to the reference image, 2D reference image is converted to 3D image and then generate images from the different aspects of the reference image. By comparing the generated images and the current image, person re-identification can be done.

The target person can be designated from a surveillance camera image as a reference. Then many images from different aspects of the target person can be generated with 2d-to-3d image conversion method. After that, the current images are acquired with the surveillance camera and are compared to the generated difference aspect images. Thus, the target person can be identified with a feature matching method between the generated image features and the current image features. This is a totally new idea of re-identification.

In the next section, related research works are to be reviewed, followed by the proposed method. Then, experiments with a small number of samples are described. After that conclusion is described together with some discussions.

II. Related Research Works

Below are some proposed methods to prevent person re-identification and their sources.

1) Data anonymization techniques: K-anonymity: A technique to ensure that each record in a dataset is indistinguishable from the records of at least k individuals.

A model for protecting privacy is discussed as L-diversity which is an extension of K-anonymity that ensures the diversity of attribute values within each group [1] together with L-diversity of privacy beyond k-anonymity [2]. T-closeness which ensures that the distribution of attribute values within each group is close to the overall distribution is discussed [3].

In terms of data masking and sampling, reduces the risk of re-identification by masking parts of the data or by random sampling is discussed and a quantitative comparison of disclosure control methods for microdata was made [4].

2) Differential privacy: A method of preventing personal identification by adding noise during data processing. Differential privacy is discussed [5].

3) Densification techniques: Protect data from unauthorized access by densifying it. Foundations of Cryptography is published from Cambridge University Press [6].

4) Privacy policies and guidelines: Establish clear policies and guidelines for collecting, using, and sharing data. European Union General Data Protection Regulation (GDPR) - Article 25 (Data protection by design and by default) is delivered [7].

5) Data aggregation and summarization: Providing aggregated or aggregated data rather than individual data prevents personal identification. Data security is well discussed in ACM Computing Surveys [8].

6) Handcrafted feature-based methods: Early ReID research used handcrafted features such as color histograms, textures, and local features to capture human features. Viewpoint invariant pedestrian recognition with an ensemble of localized features is attempted [9]. Also, person re-identification by symmetry-driven accumulation of local features is proposed [10].

7) Metric learning-based methods: A method for learning a distance metric was proposed to appropriately calculate the similarity between people. Large scale metric learning from equivalence constraints is conducted [11]. Person re-identification by local maximal occurrence representation and metric learning is proposed and tested [12].

8) Deep learning-based methods: The method of using CNN (convolutional neural network) to simultaneously learn feature extraction and classification has become mainstream. An Improved Deep Learning Architecture for Person Re-Identification is proposed [13]. Person re-identification in the wild is attempted [14]."

9) Partial feature (part-based) method: Improves recognition performance by dividing the human body into parts and utilizing information from each part. Person retrieval with refined part pooling and a strong convolutional baseline is proposed [15]. Deeply learned part-aligned representations for person re-identification are also proposed [16].

10)Attention mechanism: Improves recognition accuracy by focusing attention on important features and areas. Attention-aware compositional network for person re-identification is proposed [17]. A dual attention matching network for context-aware feature sequence-based person re-identification is also proposed [18].

11)Methods using generative models (GAN): Through data augmentation and style conversion, the amount of training data can be increased and improve the generalization performance of the model. Unlabeled samples generated by GAN improve the person re-identification baseline in vitro is proposed [19]. Person transfer GAN to bridge domain gap for person re-identification is also proposed [20].

12)Transformer-based methods: A method has been proposed to capture long-distance dependencies using a transformer architecture. TransReID: Transformer-based Object Re-Identification is proposed relatively recently [21].

13)Video generation tools: Using tools such as KLING, DeeVid.ai, Sora, etc., different viewpoint images are generated. Then the images can be used for image matching between the reference image and the current image which results in person re-identification.

14)Major datasets: Major datasets related to person re-identification are as follows:

Market-1501[1]

DukeMTMC-reID[2]

[1] https://www.kaggle.com/datasets/pengcw1/market-1501 accessed on 18 January 2025.

[2] https://www.kaggle.com/datasets/whurobin/dukemtmcreid accessed on 18 January 2025.

CUHK03[3]

MSMT17[4]

These methods are widely used to protect data privacy and reduce the risk of person re-identification. Each method has its own advantages and limitations, so it is important to choose the method that best suits particular application.

15)Evaluation methods for re-identification accuracy: Also, there are some evaluation measures for accuracy measurements of person re-identifications as follows:

mAP (mean Average Precision)

CMC (Cumulative Matching Characteristics)

Rank-1/5/10 accuracy

These related papers and datasets are referred to as follows:

CVPR (IEEE Conference on Computer Vision and Pattern Recognition)

ICCV (IEEE International Conference on Computer Vision)

ECCV (European Conference on Computer Vision)

TPAMI (IEEE Transactions on Pattern Analysis and Machine Intelligence)

III. Proposed Methods and Systems

A. Process Flow of the Proposed Person Re-Identification

The process flow of the proposed person re-identification method is illustrated in Fig. 1. The method comprises several key steps designed to accurately re-identify a person of interest captured by surveillance cameras, even when images are taken from different viewpoints.

Step 1: Detection of the Person of Interest

Reference Image Acquisition: Initially, a person of interest is selected from images acquired by surveillance cameras based on YOLO11. This image serves as the reference image for subsequent matching processes. YOLO11 provides precise bounding boxes, adjusting the center location, size, and inclination angle to accurately localize the person.

Step 2: Generation of Multi-Angle Synthetic Images

Rotated Image Generation: From each 3D model (reference image), a set of synthetic images by rotating the model at fixed intervals is generated. Specifically, 36 images are generated for each model, rotating the model by 10 degrees per image to cover a full 360-degree rotation.

Step 4: Image Matching and Re-identification

Feature Matching: Using feature matching method, the best matched person can be selected through comparison between the generated images from the different aspects and the current image.

Cross-Checking: To enhance reliability, a cross-check can be done by finding the image with the smallest difference between the current image and the rotated images generated from its own 3D model. This step validates the consistency of the current image's representation.

Optimization for computational efficiency is performed as follows:

Utilization of 2D Image Differences: To reduce computational resources, differences directly between the acquired 2D images and the generated synthetic 2D images are computed, instead of comparing complex 3D models or higher-dimensional feature representations. This approach simplifies computations while maintaining accuracy.

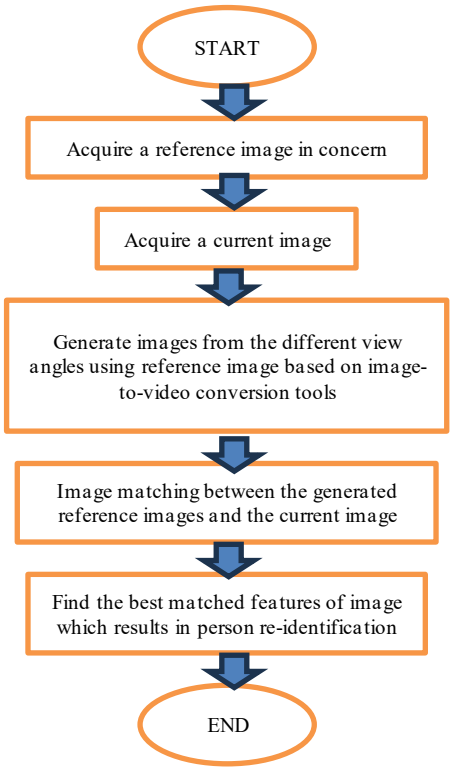


Fig. 1. Process flow of the proposed person re-identification.

Originality and advantages of the proposed method are as follows:

The key innovation of our person re-identification method lies in the integration of 3D reconstruction and multi-angle synthetic image generation to address appearance variations due to different viewpoints. By converting 2D images into 3D models and generating images from multiple angles, this method can effectively match images taken from different viewpoints, overcoming one of the primary challenges in person re-identification.

Addressing Appearance Variations: This method compensates for changes in appearance caused by different shooting angles, lighting conditions, and partial occlusions.

[3] https://www.kaggle.com/datasets/priyanagda/cuhk03 accessed on 18 January 2025.

[4] https://paperswithcode.com/dataset/msmt17 accessed on 18 January 2025.

Reducing Ambiguity Among Similar Individuals: By analyzing multi-angle representations, the method enhances discrimination between individuals who may appear similar in traditional 2D images.

Efficiency in Real-World Applications: The approach balances computational efficiency with accuracy, making it suitable for practical applications in surveillance and security systems.

B. Similarity Measure

Another key issue here is how to evaluate similarity between the 2D images in concern and the 2D images converted from the image-to-video conversion method. The method for evaluating the similarity between images should be selected taking into consideration the purpose, image features, calculation costs, etc. Here, it will be introduced some representative methods along with references.

1) Comparison at the pixel level:

a) Euclidean distance: The RGB values of each pixel are treated as vectors, and the Euclidean distance between two image vectors is calculated. The smaller the distance, the higher the similarity [22].

- Advantages: Simple and fast calculation.
- Disadvantages: Weak to translation, rotation, and scale changes of images.

b) Manhattan distance: Calculates the sum of the absolute value of the difference between the RGB values of each pixel. As with Euclidean distance, the smaller the distance, the higher the similarity [23].

- Advantages: Less susceptible to outliers than Euclidean distance.
- Disadvantages: As with Euclidean distance, it is weak to translate, rotation, and scale changes of images.

c) Histogram comparison: This method compares the brightness of histograms of images. The similarity of images can be evaluated by calculating the distance between histograms [24].

- Advantages: Robust to changes in image brightness.
- Disadvantages: Since it does not consider the content or shape of the image, histograms may be similar even for different images.

2) Feature-based comparison

a) SIFT (Scale-Invariant Feature Transform)[5]: Extracts features that are invariant to scale and rotation and evaluates the similarity between images by matching.

b) SURF (Speeded Up Robust Features)[6]: A method that can extract features faster than SIFT [25].

c) ORB (Oriented FAST and Rotated BRIEF)[7]: A feature extraction method that is even faster than SIFT and SURF [26].

d) CNN (Convolutional Neural Network)[8]: Extracts features from images using deep learning and evaluates the similarity between images by comparing them [27].

- Advantages: Highly accurate similarity evaluation is possible.
- Disadvantages: Depends on the quality of the training data. High computational cost.

3) Structural Similarity Index Measure (SSIM)[9]

a) SSIM (Structural Similarity Index Measure): A similar index that considers the characteristics of human vision. Similarity is evaluated based on three elements: image brightness, contrast, and structure [28].

- Advantages: Evaluation close to human vision is possible.
- Disadvantages: High computational cost.

4) Other

a) Earth Mover's Distance (EMD)[10]: An index that measures the distance between two distributions and is used for comparing image histograms [29].

b) Hausdorff Distance: An index that measures the distance between two-point sets and is used for comparing image shapes [30].

IMGSIM[11] is very common for image similarity measures.

In this study, just feature matching methods of ORB and SURF [31] are used for image similarity measurements.

IV. Experiments

A. Extraction of the Person in Concern from the Acquired 2D Images

For this experiment, a subset of the Market-1501 dataset, a well-known source for person re-identification, was utilized. Fig. 2(a) displays a selection of these images used in the study. From this dataset, two specific images of the individual of interest—referred to as the reference and current images—were extracted, as shown in Fig. 2(b) and (c). The images capture both the front and side views of the subject. Also, another example of a reference and a current image are shown in Fig. 2(d) and (e). These two cases are just test examples for validation of the proposed person re-identification method.

[5] https://en.wikipedia.org/wiki/Scale-invariant_feature_transform accessed on 18 January 2025.

[6] https://en.wikipedia.org/wiki/Speeded_up_robust_features accessed on 18 January 2025.

[7] https://en.wikipedia.org/wiki/Oriented_FAST_and_rotated_BRIEF accessed on 18 January 2025.

[8] https://en.wikipedia.org/wiki/Convolutional_neural_network accessed on 18 January 2025.

[9] https://en.wikipedia.org/wiki/Structural_similarity_index_measure accessed on 18 January 2025.

[10] https://en.wikipedia.org/wiki/Earth_mover%27s_distance accessed on 18 January 2025.

[11] https://github.com/totogot/ImageSimilarity/blob/main/src/ImgSim/image_similarity.py accessed on 18 January 2025.



(a) Portion of images in the Market-1501



(b) Reference image (Case1)



(c) Current image of person in concern (Case1)



(d) Another reference image (Case 2)



(e) Another current image (Case 2)

Fig. 2. Portion of Market-1501 and the extracted person in concern from the image dataset.

B. Image-to-Video Conversion Using KLING as well as DeeVid.ai

Just two examples of the resultant images obtained from the image-to-video conversion tools of KLING and DeeVid.ai are shown in Fig. 3. For these cases, the resultant images are 30 degrees rotated in horizontal direction from the original reference image.

The spatial resolution of the resultant images is superior to the reference images. It seems that the image-to-video conversion tools make models of targets and create the rotated images. Therefore, it is possible to create any rotated target image with arbitrary rotation angles. Thus, it is easier to extract features form the images for feature matching.



(a) Reference image (Case 2)



(b) 30 degrees rotated image (Case 2)



(c) Reference image (Case 1)



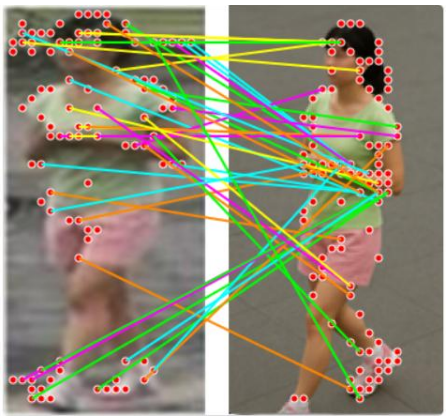
(d) 30-degree rotated image (Case 1)

Fig. 3. Examples of the resultant images obtained from the image-to-video conversion tools of KLING and DeeVid.ai

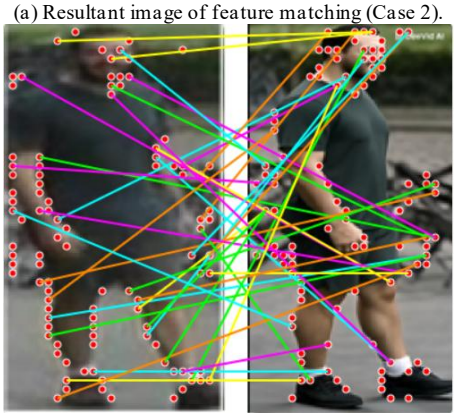
C. Feature Matching Between the Reference and the Rotated Images

ORB (Oriented FAST and Rotated BRIEF) feature point detection and matching are performed between two images. SURF is also used for confirmation of feature matching performance (Fig. 4). Feature points are detected in images (corners, edges, texture change points, etc.) and correspondences between different images are found. Red circles indicate feature points, and green lines indicate correspondence between matched feature points.

Image feature points for the reference and the 30-degree rotated images are 100. Also, the matching points between both are 31 and 42 for Case 1 and 2, respectively. Therefore, if the threshold of matching success rate is set at 30%, then the proposed person re-identification makes 100% identification success rate.



(a) Resultant image of feature matching (Case 2).



(b) Resultant image of feature matching (Case 1).

Fig. 4. Results from the feature matching.

D. Overall Matching Accuracy for the Market 1501 Database

In the previous section, just two examples of the success rate of the proposed person re-identification are demonstrated with a person image as a reference, together with the rotated image by every 10 degrees using KLING as well as DeeVid.ai. The highest matching success rate for the rotation angle is adopted for the determination of the identification success rate.

The same experiments are conducted for all the images included in Market-1501 with the same threshold of a matching success rate of 30. The results show more than 80% success rate. In the latest research on person re-identification in Market-1501, methods utilizing Vision Transformer have achieved excellent recognition accuracy. This method achieves high stability and accuracy by focusing on a person's distinctive features and their interrelationships, demonstrating excellent results at Rank-1 (the highest match rate).

Compared to previous CNN-based models, Vision Transformer enables more detailed feature extraction, resulting in improved Rank-1 accuracy on the Market-1501 dataset. While the specific highest success rate (the specific value of Rank-1 accuracy) varies depending on the paper or publication, advanced methods from 2024 onward have been reported to achieve accuracies of around 80%.

As such, the success rate of person re-identification in Market-1501 is steadily improving with technological advances, and the latest Transformer-based models are currently producing excellent results.

Therefore, it can be said that the proposed method reached the highest rank in terms of person re-identification success rate.

E. Cross-Checking Between the Current Image with Different Viewing Angles and the Reference Image, and Between the Reference Image with Different Viewing Angles with the Current Image

The proposed method introduces a robust approach to person identification through a unique bidirectional comparison system. This process consists of two complementary analysis stages:

1) Primary analysis: The system first analyzes multiple viewpoints of the current subject image, comparing each against the reference image to identify the closest match. However, this unidirectional comparison alone may not provide sufficient accuracy for reliable identification.

2) Enhanced cross-checking: To strengthen the identification accuracy, the system implements a novel bidirectional verification process:

a) Current-to-reference: Multiple viewpoints of the current subject are compared against the reference image

b) Reference-to-current: Multiple viewpoints of the reference image are compared against the current subject

This innovative bidirectional cross-checking methodology significantly enhances the reliability of person identification by:

Reducing false positives through dual verification.

Compensating for variations in viewing angles.

Providing a more comprehensive similarity assessment.

This unique approach to cross-validation represents a significant advancement over traditional unidirectional comparison methods, offering a more robust and reliable framework for person identification.

V. Conclusion

A novel approach to person re-identification using image-to-video generation is proposed.

This research presents an innovative method for person re-identification that leverages image-to-video conversion technology to address fundamental challenges in privacy protection and data security. Person re-identification—the process of identifying individuals within anonymized datasets—has historically faced significant accuracy limitations.

Traditional person re-identification methods struggle with two primary challenges:

- Appearance Variations: Subject features can vary significantly based on viewing angle and environmental conditions.

- Similar Appearances: Multiple individuals in datasets may share similar visual characteristics, complicating unique identification.

This approach significantly improves identification accuracy by:

- Compensating for viewing angle variations.
- Providing more robust feature comparison.
- Enabling better distinction between visually similar individuals.

Experimental validation demonstrates the effectiveness of this image-to-video conversion-based approach, showing marked improvements in identification accuracy compared to traditional methods. The ability to generate and analyze multiple viewpoints from 3D models provides a more comprehensive and reliable basis for person re-identification.

This improved description better organizes the information, clarifies the technical concepts, and emphasizes the innovative aspects of the proposed method while maintaining all essential information from the original text.

The detailed conclusions are as follows:

Experiments conducted using this method have shown significant improvements in person re-identification accuracy.

Effectively Addressing Appearance Variations: The use of 3D models and synthetic views accounts for changes in viewpoint, posture, and partial occlusions.

Enhancing Robustness: image-to-video creation provides a more stable representation of the individual's features, less affected by transient factors like lighting and minor attire changes.

Reducing False Matches: Improved discrimination between individuals helps in environments with many people who have similar appearances.

The proposed method offers a promising solution to the challenges faced in person re-identification by:

Mitigating Appearance Variations: Addressing one of the core issues affecting recognition rates through multi-angle analysis.

Leveraging Advanced Technologies: Utilizing cutting-edge image-to-video conversion techniques and depth estimation models enhances the robustness of the system.

Improving Practical Applicability: The method's increased accuracy brings it closer to stable use in real-world environments, benefiting applications in surveillance, security, and beyond.

VI. Future Research Works

Although it is confirmed that the proposed method does work for person re-identification with a well performance, it must be verified with the well-known databases of Market-1501, DukeMTMC-reID, CUHK03, and MSMT17. Larger-scale experiments and comparisons with state-of-the-art models (e.g. Transformer-based methods) are necessary. Also, performance comparison to the existing methods has to be made.

Continued research and development in this area can further refine the approach, potentially integrating it with other emerging technologies to overcome remaining challenges in person re-identification.

References

[1] Sweeney, L. (2002). "k-anonymity: A model for protecting privacy." International Journal on Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5), 557-570, 2002.

[2] Machanavajjhala, A., Kifer, D., Gehrke, J., & Venkitasubramaniam, M. (2006). "L-diversity: Privacy beyond k-anonymity." ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1), 1-52, 2006.

[3] Li, N., & Li, T. (2007). "t-closeness: Privacy beyond k-anonymity and l-diversity." In Proceedings of the 23rd International Conference on Data Engineering (ICDE '07), 106-115, 2007.

[4] Domingo-Ferrer, J., & Torra, V. (2002). "A quantitative comparison of disclosure control methods for microdata." In Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies (pp. 113-134), 2002.

[5] Dwork, C. (2006). "Differential privacy." In Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP '06), 1-12, 2006.

[6] Goldreich, O. (2004). Foundations of Cryptography: Volume II Basic Applications. Cambridge University Press, 2004.

[7] European Union General Data Protection Regulation, Establish clear policies and guidelines for collecting, using, and sharing data. GDPR - Article 25 (Data protection by design and by default) 2024, https://gdprhub.eu/Article_25_GDPR referred on 16 December 2024.

[8] Denning, D. E., & Denning, P. J. (1979). "Data security." ACM Computing Surveys, 11(3), 227-249, 1979.

[9] Gray and Tao (2008), "Viewpoint invariant pedestrian recognition with an ensemble of localized features" European Conference on Computer Vision (ECCV), pp. 262–275, 2008.

[10] Farenzena, M., Bazzani, L., Perina, A., Murino, V., & Cristani, M. (2010), "Person Re-identification by Symmetry-Driven Accumulation of Local Features", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2360–2367, 2010.

[11] Köstinger et al. (2012), "Large Scale Metric Learning from Equivalence Constraints" IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2288–2295, 2012.

[12] Liao, S., Hu, Y., Zhu, X., & Li, S. Z. (2015), "Person Re-identification by Local Maximal Occurrence Representation and Metric Learning", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2197–2206, 2015.

[13] Ahmed, E., Jones, M., & Marks, T. K. (2015), "An Improved Deep Learning Architecture for Person Re-Identification", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3908–3916, 2015.

[14] Zheng, Z., Zheng, L., & Yang, Y. (2016), "Person Re-identification in the Wild", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3346–3355, 2016.

[15] Sun, Y., Zheng, L., Yang, Y., Tian, Q., & Wang, S. (2018), "Beyond Part Models: Person Retrieval with Refined Part Pooling (and a Strong Convolutional Baseline)", European Conference on Computer Vision (ECCV), pp. 480–496, 2018.

[16] Zhao, H., Tian, M., Sun, S., Shao, J., Yan, J., Yi, S., Wang, X., & Tang, X. (2017), "Deeply-Learned Part-Aligned Representations for Person Re-Identification", IEEE International Conference on Computer Vision (ICCV), pp. 3219–3228, 2017.

[17] Xu, J., Zhao, R., Zhu, F., Wang, H., & Ouyang, W. (2018), "Attention-Aware Compositional Network for Person Re-Identification", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2119–2128, 2018.

[18] Si, J., Zhang, H., Li, C., Zhao, X., & Tian, Q. (2018), "Dual Attention Matching Network for Context-Aware Feature Sequence based Person Re-Identification", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5363–5372, 2018.

[19] Zheng, Z., Zheng, L., & Yang, Y. (2017), "Unlabeled Samples Generated by GAN Improve the Person Re-Identification Baseline in vitro", IEEE International Conference on Computer Vision (ICCV), pp. 3754–3762, 2017.

[20] Wei, L., Zhang, S., Gao, W., & Tian, Q. (2018), "Person Transfer GAN to Bridge Domain Gap for Person Re-Identification", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 79–88, 2018.

[21] He, S., Luo, H., Wang, P., Wang, F., Li, H., & Jiang, W. (2021), "TransReID: Transformer-based Object Re-Identification", IEEE International Conference on Computer Vision (ICCV), pp. 15013–15022, 2021.

[22] Gonzalez, R. C., & Woods, R. E. (2002). Digital Image Processing (2nd ed.). Prentice Hall, 2002.

[23] Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media, Inc., 2008.

[24] Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 60(2), 91–110, 2004.

[25] Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In European Conference on Computer Vision (pp. 404-417). Springer, Berlin, Heidelberg, 2006.

[26] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2564-2571), 2011.

[27] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (pp. 1097-1105) 2011.

[28] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4), 600-612, 2004.

[29] Rubner, Y., Tomasi, C., & Guibas, L. J. (2000). The Earth Mover's Distance as a Metric for Image Retrieval. International Journal of Computer Vision, 40(2), 99–121, 2000.

[30] Huttenlocher, D. P., Klanderman, G. A., & Rucklidge, W. J. (1993). Comparing images using the Hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(9), 850-863, 1993.

[31] Kohei Arai, Indra Nugraha Abdullah, Kensuke Kubo, Katsumi Sugawa, Methods for wild pig identifications from moving picture and discrimination of female wild pigs based on feature matching method, International Journal of Advanced Research on Artificial Intelligence, 4, 7, 41-46, 2015.

AUTHOR'S PROFILE

Kohei Arai, received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science and Technology of the University of Tokyo from April 1974 to December 1978 was also with the National Space Development Agency of Japan from January 1979 to March 1990. From 1985 to 1987, he was with the Canada Centre for Remote Sensing as a Post-Doctoral Fellow of the National Science and Engineering Research Council of Canada. He moved to Saga University as a Professor in the Department of Information Science in April 1990. He was a counselor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology from 1998 to 2000. He was a councilor of Saga University in 2002 and 2003. He was also an executive councilor for the Remote Sensing Society of Japan from 2003 to 2005. He has been a Science Council of Japan Special Member since 2012. He is an Adjunct Professor at Brawijaya University. He is also an Award Committee member of ICSU/COSPAR. He is also an adjunct professor of Nishi-Kyushu University and the Kurume Institute of Technology Applied AI Research Laboratory. He wrote 129 books and published 748 journal papers as well as 569 conference papers. He received 98 awards, including the ICSU/COSPAR Vikram Sarabhai Medal in 2016, and the Science Award of the Ministry of Mister of Education of Japan in 2015. He is now Editor-in-Chief of IJACSA and IJISA. http://teagis.ip.is.saga-u.ac.jp/index.html

Design of a Modular Architecture Based on AI and Blockchain for Personalized Microcredits Using Open Finance

Pedro Hidalgo, Ciro Rodríguez, Luis Bravo, Cesar Angulo
Faculty of Systems Engineering and Computer Science, Universidad Nacional Mayor de San Marcos, Lima – Perú

***Abstract*—This paper presents the design and validation of a modular architecture for smart microcredits, aimed at expanding credit access for populations excluded from the traditional financial system. The solution integrates three key technological components: data acquisition through Open Finance, automated risk assessment using Artificial Intelligence (AI) models, and the execution of smart contracts on blockchain. A functional prototype was developed to process applications manually submitted by users without prior financial history, utilizing a LightGBM model trained on real, anonymized data. The model was integrated into the system workflow to generate automatic credit conditions and register decisions on the blockchain without direct human intervention. During the validation phase, the model achieved an Area Under the Curve (AUC) of 0.94, supporting its discriminative power within the automated flow. The overall technical validation demonstrates the feasibility of offering personalized, traceable, and secure credit services through open and decentralized technologies. The use of alternative unstructured data, as well as the expansion into production environments, is proposed as a future line of development. In our system, Open Finance provides consented financial data off-chain; the ML model estimates default probability and outputs an eligibility decision; a rule engine maps the score to personalized loan terms; and blockchain smart contracts only record loan terms and execution events on-chain (no personal data). This separation ensures auditability (on-chain) and privacy (off-chain).**

Keywords*—Smart microcredits; artificial intelligence; open finance; blockchain; smart contracts; financial inclusion

I. Introduction

Access to formal credit remains a significant barrier for millions of people in Latin America and other regions of the world [1][2], particularly for informal workers, young individuals without a credit history, and small entrepreneurs. Despite the advancement in the digitalization of financial services [2][3], traditional credit scoring models and banking intermediation continue to exclude large segments of the population [4][2], perpetuating cycles of informality and inequality [5].

In response to this challenge, emerging technologies offer a concrete opportunity to redesign financial services from a more inclusive, automated, and adaptive perspective [1][3][6]. In particular, the convergence of Open Finance [3], Artificial Intelligence (AI) [4][7], and blockchain technology [8][9] enables the conception and development of new architectures that decentralize decision-making, reduce operational costs, and personalize financial products according to the applicant's real characteristics [11].

Recent advances create a concrete opportunity to redesign this workflow end-to-end. Open Finance enables consented access to user financial data; machine learning (ML) can estimate default risk from structured inputs; and blockchain offers programmable execution and immutable audit trails. Yet, most existing efforts treat these components in isolation (AI-only scoring, sandbox blockchain pilots, or partial integrations) and rarely present a modular, low-cost, and implementation-oriented architecture suitable for constrained environments.

This paper introduces and validates a modular architecture for smart microcredits that combines consented data ingestion (off-chain), ML-based risk estimation, policy-driven personalization of loan terms, and on-chain recording of loan execution events. The approach emphasizes separation of concerns (prediction vs. decisioning vs. auditability), privacy by design (no personally identifiable information on-chain), and progressive deployment in both urban and low-connectivity contexts. It is inspired by validated software engineering practices [12] and supported by open-source technologies [13][14]. The goal is not to introduce a new financial product concept, but rather to offer a feasible, scalable, and adaptable technical solution [17] that can be implemented in real-world settings with reduced costs and significant social impact [1].

Scope and Roles. In this work, personalized microcredits are loans whose amount, rate and term are automatically tailored to each applicant based on an ML-derived risk score and policy rules. Open Finance acts as a consented data source (off-chain). The ML model estimates default risk from structured features and returns a score/label. A decisioning rule engine converts that output into loan terms (personalization). Finally, blockchain executes and immutably records contract terms and events (hashes/IDs only, no PII on-chain). This explicit separation clarifies responsibilities: ML = prediction, rule engine = product personalization, blockchain = execution + auditability.

Additionally, this study includes a comparative analysis of transaction costs (gas fees) and latency across different blockchain networks (Ethereum Sepolia, Polygon Mumbai, Gnosis Chain), aiming to assess the economic feasibility and scalability of the proposed architecture in production environments [34][35]. The results of this analysis are presented in Section VI.

From a technical standpoint, this proposal is distinguished by its modular design [12], interoperability capabilities [13], and the potential for progressive deployment [14][15]. It targets developers, financial institutions, fintechs, and cooperation entities seeking to globalize technology-based solutions aimed at sustainable financial inclusion [1][17].

Research questions. We investigate the following: RQ1: Can a modular architecture that separates ML prediction, policy-based personalization, and on-chain execution be implemented end-to-end with low operational cost? RQ2: Does recording only non-PII artifacts on-chain (hashes/IDs) preserve auditability while respecting privacy-by-design? RQ3: Under realistic constraints (low connectivity, low-end devices), is the approach operationally viable in terms of latency, traceability, and maintainability?

Organization of the paper. The remainder of the paper is organized as follows. Section II reviews related work at the intersection of microcredit, Open Finance, ML-based credit scoring, and blockchain. Section III describes the methodology and development process. Section IV details the proposed architecture and its modules. Section V presents application scenarios that illustrate the end-to-end flow. Section VI reports prototype validation results, including a comparison of gas fees and latencies across networks. Section VII discusses implications, limitations, and ethical/regulatory considerations. Section VIII concludes and outlines future work. Appendix A provides implementation details of the functional prototype (codebase, data flow, and artifacts).

II. Related Work

The literature and prior projects in this domain can be grouped into four main categories:

1) Digital microfinance with AI: AI-based Credit Scoring Models in Microfinance (2025) [18] analyzes how AI models enable the assessment of loan applicants without banking history by leveraging alternative data sources such as mobile consumption and social media activity. It highlights benefits such as broader coverage and risk reduction, while also noting challenges related to privacy and algorithmic bias.

2) Efficient credit scoring: Credit Scoring for Good (2020) [19] demonstrates how mobile applications use machine learning and feature engineering to improve financial inclusion through efficient predictive scoring models.

3) AI + Blockchain integration for risk assessment: Blockchain-Based Deep Learning Model LSTM-X (2024) [20] proposes a system in which an LSTM model analyzes financial time series, while blockchain ensures data integrity and decentralized institutional access—enhancing both accuracy and trust.

Robust Integration of Blockchain and Explainable Federated Learning for Automated Credit Scoring (2024) [22] combines blockchain with federated learning to enable decentralized credit scoring models, emphasizing user privacy.

Hybrid Blockchain + XAI for Credit Scoring (2024) [32] focuses on automating credit decisions in a reliable and auditable manner using explainable artificial intelligence (XAI).

4) Open finance / open banking + credit scoring: A recent MDPI publication shows that transaction data obtained through Open Banking APIs, when combined with deep learning techniques, can outperform conventional models used by a Norwegian bank in predicting defaults, thus closing the gap for new customers [23].

Additionally, the synergy between AI and Open Banking is acknowledged as a key enabler for delivering personalized and secure financial services through fintech solutions.

TABLE I Identified Gaps in Current Literature on Emerging Technologies Applied to Microcredit

Incomplete Approaches	Many studies implement AI or blockchain separately or within institutional settings, but do not integrate them into solutions for low-value microcredits.
Lack of Concrete Modular Architecture	There are no proposals that integrate AI, Open Finance, and smart contracts into a technical architecture designed for progressive deployment.
Limited Focus on Real-World Implementation	Most existing studies present theoretical models or proof-of-concept prototypes, without offering a practical path for developers or direct implementation.
Absence of Blockchain Operational Cost Analysis	Existing works lack comparative analyses of gas fees and latency across different blockchain networks—an essential factor in assessing the economic feasibility of such solutions.

5) Comparative analysis of transaction costs and latency in low-cost blockchain networks: Pérez and Santos (2025) [34] present a study of gas fees on Ethereum Sepolia, Polygon (Mumbai), and Gnosis Safe, showing savings of up to 80% in transaction costs and latencies ranging from 4 to 10 seconds on Layer 2 testnets.

Ivanov et al. (2025) [35] confirm these findings and discuss how these parameters affect the economic feasibility of large-scale deployments.

In summary, as outlined in Table I, existing studies have made progress in various directions, yet still lack an integrated, modular, and development-oriented solution. The following section introduces an architecture designed to address these gaps through an applied and replicable approach.

Advantages over prior approaches. Unlike prior work that treats ML, Open Finance, and blockchain in isolation, our system: 1) separates concerns (prediction vs. rule-based personalization vs. on-chain execution); 2) keeps no PII on-chain while preserving auditability; 3) provides implementation-ready modules (UI/API/model/contracts); 4) reports operational evidence (gas/latency) to inform feasibility; and 5) targets low-connectivity scenarios via lightweight clients and asynchronous signing.

Comparative positioning with prior work. To enable a like-for-like comparison, Table II summarizes representative studies across seven dimensions: data source (Open Finance vs. forms/alt-data), ML technique, the role of blockchain, product personalization, on-chain privacy, cost/latency analysis, and

implementation maturity. This complements Table I (identified gaps) with an operational view of scope and deliverables.

III. Methodology

We evaluate the prototype on the Loan Approval Prediction dataset [37]. Basic demographics, income, employment, and loan attributes are included.

Dataset and preprocessing. We use the Kaggle Loan Approval Prediction dataset [37]. We select structured features (demographics, income, employment, housing, and loan attributes), normalize types, and handle missing values via simple imputation (median/mode). Categorical variables are one-hot encoded; numerical features are standardized only when required by the model. We split data into train/validation/test with stratification on the target and handle class imbalance through model-level weighting (e.g. scale_pos_weight). All steps are scripted and versioned to enable reproducibility (see Appendix A and [36]).

Fig. 1 illustrates our project-specific development process: a Waterfall backbone with six phases (A–F), complemented by 2-week Agile MLOps sprints for model training/validation between Design and Validation. This adaptation preserves clear stage-gates and deliverables while enabling rapid iteration on the ML component. As shown, project activities are organized into sequential phases, where the start of each stage depends on the successful completion of the previous one. This methodology provided structure, clarity in planning, and strict control over the development process, making it suitable for projects with clearly defined objectives, such as the construction of a functional prototype for smart microcredits based on emerging technologies.

While the Waterfall model offers clear deliverables and rigorous control at each stage, we suggest complementing this approach with Agile development sprints for the MLOps component. After completing the initial design phase, each iteration of model training and validation can be managed in 2-week sprints, allowing for rapid adjustments to data and parameters before proceeding to the next Waterfall phase.

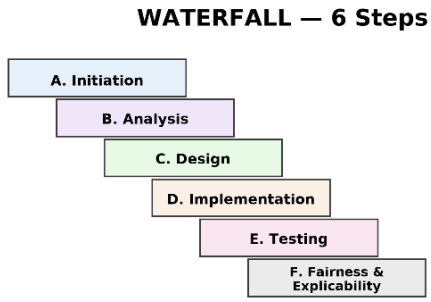


Fig. 1. Project-specific development process: Waterfall phases (A–F) with agile MLOps sprints (2-week cycles) for model iteration.

A. Phase 1 – Initiation

1) Project scope and objective definition: In this initial phase, both functional and non-functional objectives were established, thereby delimiting the scope of the proposed solution. The main goal was defined as the development of a Minimum Viable Prototype (MVP) aimed at managing digital microcredits with high potential for scalability, security, and reproducibility.

TABLE II Comparative Positioning Against Prior Work (✓ = Explicitly Addressed; — = Not Addressed; ? = Unclear)

+Work (ref.)	Open Finance data	ML technique	Blockchain role	Personalized loan terms	PII stored on-chain	Gas/latency analysis	Implementation maturity
[18] AI-based Credit Scoring in Microfinance (2025)	— (alt-data)	AI scoring (various)	—	—	—	—	Conceptual / partial.
[23] Open Banking + Deep Learning (MDPI)	**✓ (OB APIs)**	Deep learning	—	—	—	—	Empirical scoring; no chain.
[20] LSTM-X + Blockchain (2024)	—	LSTM	**Integrity / decentralized access control**	—	?	—	Prototype; not microcredit-specific.
[22] Federated Learning + Blockchain (2024)	—	**Federated learning + XAI**	**Decentralized model mgmt / privacy**	—	**No** (privacy-oriented)	—	Method paradigm; no product terms.
[32] XAI + Blockchain for Credit Modeling (2024)	—	XAI over ML	**Auditability / secure modeling**	—	?	—	Automation focus; no OB.
This work (ours)	**✓ (consented Open Finance) + form**	**LightGBM**	**Execution + audit trail**	**✓ (policy-driven)**	**No** (hashes/IDs only)	**✓ (Section VI)**	**MVP**: end-to-end UI/API/model/contracts.

Notes. Personalization = mapping ML risk (PD) → amount/APR/term via rules (Section IV). On-chain PII = whether personally identifiable data is written to the ledger (we store only refs/hashes). Implementation maturity reflects reported artifacts (code, UI, contracts). See the cited works in Section II and our prototype details in Appendix A.

2) Project Deliverables

- Scope document and specification of functional and technical requirements.
- Analysis of the microcredit life cycle and identification of technological intervention points.
- Design of a modular architecture based on Open Finance, Artificial Intelligence, and blockchain.
- Development of a minimum viable prototype with automated scoring and smart contract deployment on testnet.
- Technical validation of the prototype through controlled tests on performance, traceability, and security.
- Recommendations for scalability and replication in digital financial inclusion contexts.

B. Phase 2 – Analysis

1) Microcredit lifecycle analysis: An in-depth analysis was conducted on the complete lifecycle, from credit issuance to repayment, structured into six key phases:

- Data Collection: This phase involves gathering essential financial information from alternative sources, as well as demographic and behavioral data of the applicant. However, behavioral data has not yet been incorporated into the current implementation.
- Risk Assessment: Processing and evaluation through machine learning models.
- Loan Condition Generation: Automation of loan amount, interest rates, terms, and guarantees.
- Contract Execution: Formalization through smart contracts deployed on blockchain.
- Monitoring: Tracking of repayment compliance and financial behavior (not implemented in this phase).
- Closure: Conclusion of the cycle, including result logging and feedback generation.

This analysis helped identify areas suitable for automation of repetitive tasks, enhancement of traceability, and support for decision-making through ML algorithms. It also supported the definition of functional requirements for each module of the system.

2) Risk and constraint identification: This phase also involved the identification of technical, social, and regulatory limitations, such as:

- Limited connectivity in rural areas.
- Low levels of digital literacy.
- Need for interoperability with existing financial systems.

Mitigation strategies were proposed, including the use of lightweight mobile platforms, user-friendly interfaces, and compliance with data protection regulations.

C. Phase 3 – Design

1) Technology selection: The solution is built upon three technological pillars:

- Open Finance: Access to financial data through standardized APIs, ensuring consent and traceability.
- Artificial Intelligence: Use of predictive models such as LightGBM and neural networks to assess credit risk.
- Blockchain: Deployment of smart contracts to securely and transparently execute loan conditions.

The selection criteria included scalability, low cost, open-source adoption, and cross-platform compatibility.

2) Modular architecture design: The proposed architecture is based on a modular and decoupled structure, organized into three main layers that interact through web services (APIs) and smart contracts. Fig. 2 presents the layered modular architecture used throughout Section IV:

- Data Input Layer:

A web or mobile client submits a loan request to the backend via FastAPI. This web API initially validates the submitted information and manages the business logic required for credit evaluation. If the applicant's profile requires assessment, the scoring process is initiated.

- Machine Learning Layer:

The backend forwards the request to a pre-trained Machine Learning model (LightGBM), which evaluates the credit profile. If the model approves the loan, a blockchain transaction is triggered. This involves interaction with a pre-deployed smart contract on the Ethereum Sepolia testnet, developed using Remix IDE and the Solidity programming language.

- History and Logging Layer:

Once the loan is approved:

1) Key credit terms and applicant reference ID are collected; a transaction is built to call the smart contract (to/from, value, gas, calldata), the user signs in MetaMask, and the resulting transaction hash is returned and stored off-chain.

2) A personalized transaction is built that calls a smart contract function, encoding required parameters in hexadecimal format.

3) Technical parameters are configured: contract address (to), sender address (from), value in Wei, estimated gas, and gasPrice.

4) The transaction is presented to the user via MetaMask for digital signing.

5) Upon confirmation, the transaction is sent to the Ethereum network, generating a unique transaction hash and linking it to the smart contract address.

6) Finally, the system returns the transaction hash to the frontend, confirming that the loan has been transparently and immutably recorded on the blockchain.

This process ensures not only traceability of the automated decision but also the integrity and public verifiability of the credit record, aligning the system with principles of transparency, security, and auditability.

Finally, this layer stores the history of loan requests, scoring results, and key transaction data in a PostgreSQL database, thereby ensuring traceability and information persistence.

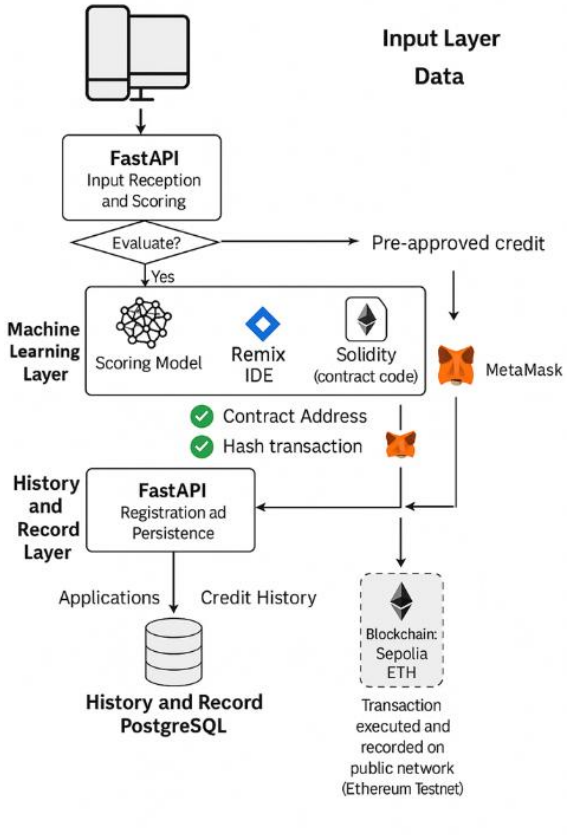


Fig. 2. Proposed modular architecture.

This design enables the decoupling of input, evaluation, and persistence processes, thereby facilitating scalability, integration with MLOps tools in subsequent phases, and the secure incorporation of blockchain for smart contracts.

D. Phase 4 – Implementation

For a detailed technical implementation, see *Appendix A*.

1) Development of a functional prototype: During this phase, a minimum viable prototype was developed, initially integrating the following three modules:

- Data entry via a simulated form using a REST API.
- An AI model trained on an anonymized real-world dataset.
- A credit condition generation engine.

Additionally, a basic smart contract was deployed in Solidity on a blockchain test network (Ethereum testnet), to validate the contractual flow and its traceability. The full MVP codebase, including API, model service and smart-contract artifacts, is publicly available [36].

2) Development environment: The prototype was built using open-source tools and frameworks selected for their lightweight nature, ease of integration, and suitability for academic or local testing environments:

- Python as the main programming language, with libraries such as Scikit-learn and LightGBM for building and testing the credit risk model.
- FastAPI to create a lightweight REST API that serves as backend and orchestrates business logic.
- Solidity for writing the smart contract, deployed using Remix IDE on the Sepolia testnet.
- MetaMask as a tool for signing blockchain transactions in the test environment.
- PostgreSQL as a unified database to store loan applications, scoring results, and cross-references to deployed contracts.
- Git for version control of the prototype.
- Docker was considered for reproducible deployment environments, although it was not used in this initial iteration.

The diagrams detailing the technological architecture, functional components, and operational flow are included in this work (Fig. 3 depicts the technology stack and deployment workflow that implements these modules) as part of the technical documentation supporting the feasibility of the proposed approach.

E. Phase 5 – Validation

1) Technical evaluation: The prototype was validated through functionality and performance tests. Test scenarios were designed to simulate real-world microcredit cases. The following aspects were evaluated:

- System response time
- Accuracy of the credit risk model
- Contract security robustness
- Adaptability for users with low digital literacy

2) Results obtained: The system demonstrated appropriate processing times, transparent traceability of credit decisions, and the ability to adapt to varying rules based on user profiles. Areas for improvement were identified and will be addressed in future phases.

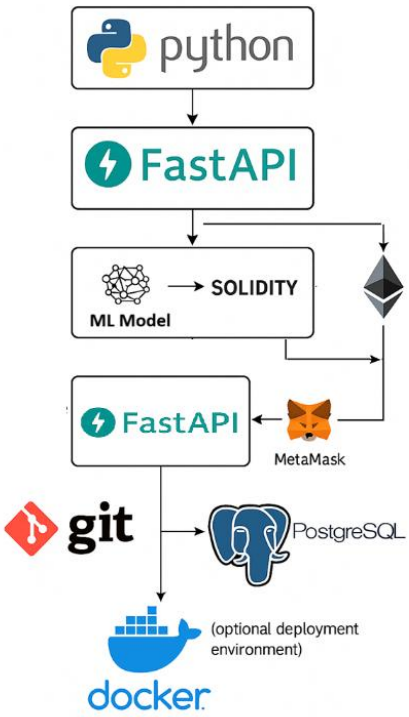


Fig. 3. Technology architecture.

F. Fairness and Explainability (XAI)

Scope. Although our prototype targets architectural feasibility, we include fairness and explainability practices to reduce risk and support audits.

Metrics. We monitor demographic parity difference, equal opportunity difference, and calibration by subgroup (when protected attributes are available).

Procedure. 1) compute group metrics on validation/test; 2) if gaps exceed policy thresholds, adjust thresholds (τ_1, τ_2) per group or apply post-processing (e.g. reject option, score banding); 3) re-evaluate utility–fairness trade-offs.

Explainability. We use feature importance at model level and local attributions (e.g. SHAP) to explain decisions in manual review.

Logging & governance. Fairness/XAI artifacts (metrics, plots, explanations) are versioned alongside the model to support audits and continuous improvement.

G. Phase 6 – Final Considerations

To ensure the viability of the system in real-world environments with limited resources, several design and development strategies must be adopted:

- Implement an API-first approach along with a modular microservices-based architecture [12]. This approach allows each system component to function independently, facilitating decoupled development, testing, deployment, and maintenance. The microservices architecture is adopted due to its high scalability—each module can scale independently based on demand—and its modularity, which allows functionalities to be integrated or replaced without affecting the rest of the system. This flexibility is key to adapting to changing contexts, incorporating new technologies, or complying with local regulations without redesigning the entire solution.
- Establish a clear separation between model training—conducted in a controlled environment—and its use in production via secure endpoints.
- Prioritize cross-platform compatibility, ensuring functionality even on low-end mobile devices.
- Incorporate security principles from the early design stages, including data encryption, token-based authentication, and robust session handling. (Not implemented in this version.)

These decisions not only consolidate the technical feasibility of the system but also make it a replicable, scalable, and adaptable solution for diverse scenarios—especially those requiring urgent responses in the field of financial inclusion.

TABLE III RESPONSIBILITY MATRIX OF THE PROPOSED SYSTEM

Component	Primary role	Inputs	Outputs	Persistence	On-chain footprint	Privacy boundary
Open Finance Gateway	Consented data ingestion	Bank/wallet APIs, user consent	Normalized features	PostgreSQL (off-chain)	None	PII stays off-chain
ML Scoring (LightGBM)	Default risk estimation	Feature vector	PD score + class label	Model logs (off-chain)	None	No PII on-chain
Decisioning / Rule Engine	**Personalization** of loan terms	PD score, policy params, income ratios	Amount, APR, term, conditions	Rules repo (off-chain)	None	Policy changes audited off-chain
Smart Contracts	**Execution & auditability**	Loan terms (refs/IDs), events	Immutable record, events	Blockchain	Tx hash, loan ref ID	Store only refs/hashes
History & Logging	Traceability	Events from all modules	Audit logs, analytics	PostgreSQL	Tx hash mirrors	Anonymized keys only

IV. PROPOSED ARCHITECTURE

The architecture is designed in response to the need to develop an intelligent microcredit system that is scalable, adaptable, and technically viable—even in resource-constrained environments [27][17]. Its design is based on a modular structure in which each component operates autonomously in terms of development, deployment, and maintenance [12][13]. This enables progressive evolution and adaptation to various regulatory frameworks or technological environments [3][25].

Table III provides the responsibility matrix that aligns components with their data flows and privacy boundaries.

Below is a description of the main modules comprising the architecture:

A. Module 1: Data Acquisition (Open Finance and Manual Input)

This module allows the acquisition of the data necessary for credit analysis. There are two main sources:

- Open Finance APIs: Access, with user consent, banking data such as account information, transaction history, and past loans.
- Manual or semi-structured input: When no affiliated banks are available, the user can input basic information (income, economic activity, expenses) and alternative data sources (mobile usage, digital payment history, etc.).

The module validates, normalizes, and stores the data in a structure used by the analysis engine.

B. Module 2: Risk Analysis with Artificial Intelligence

At this stage, the likelihood of borrower default is assessed using a previously trained machine learning model [4][7].

- Training: Performed on historical or synthetic datasets adjusted to the target population profile [7][19].
- Possible models: Decision trees, random forest, LightGBM, or simple neural networks, depending on data availability and computational capacity [7][10].
- Output: Risk score, basic interpretability (feature importance), and eligibility classification [26].

This module can be deployed as a microservice that exposes a REST API for invocation by the central system.

Personalization & Decisioning Layer. The ML model does not assign product terms; it only predicts risk. A separate rule engine maps the score into personalized loan terms using interpretable policies:

Inputs: PD score, income, loan_percent_income, employment flags.

Policy example:

- If PD $\leq \tau_1$: amount = min ($k_1 \cdot$income, cap_1), apr = baseAPR – δ_1, term $\in [m_1, M_1]$.
- If $\tau_1 <$ PD $\leq \tau_2$: tighten amount, apr = baseAPR, shorter term.
- If PD $> \tau_2$: reject or request guarantor.

Outputs: (amount, APR, term, grace, collateral_req).

Policies are versioned and auditable; they can incorporate social or public-policy parameters without retraining the model.

C. Module 3: Personalized Loan Terms Generation

Based on the risk score and other rules the system automatically determines microcredit terms, for example:

- Maximum loan amount.
- Applicable interest rate.
- Repayment period.
- Potential grace periods.

The rule engine can incorporate more complex logic, such as public policies, subsidies, or adjustable parameters by geography or gender [7][17]. However, these functionalities were not implemented in this stage of the project as they are outside the defined scope for this development phase.

D. Module 4: Smart Contract Execution

Once the user accepts the proposed loan conditions, the loan is registered via a function call on a pre-deployed smart contract that:

- Registers the credit terms immutably [8][13]
- Defines payment and penalty events [12][9]
- Enables automated monitoring and blocks new loans in the event of noncompliance [15]

It is recommended to start with a low-cost network (e.g., Polygon, Gnosis, or a private network based on Hyperledger Fabric) [24][28].

E. Module 5: Behavior Logging and Credit History

This module stores historical credit behavior data to:

- Retrain risk models [4][7]
- Offer new financial products based on compliance
- Allow external systems (fintechs, allied institutions) to query the history (with user consent) to offer additional services

This layer can interoperate with decentralized identity mechanisms to scale toward more distributed ecosystems [29][31].

F. Module 6: MLOps Considerations for the AI Module

The risk analysis module incorporates a machine learning component that requires rigorous technical management throughout its lifecycle. For this purpose, a basic MLOps (Machine Learning Operations) architecture is proposed, including the following practices:

- Offline supervised training on structured, historical, or synthetic datasets representing the target profiles. Standard techniques such as LightGBM, decision trees, or simple neural networks are applied [4][7].
- Periodic model validation using performance metrics such as accuracy, recall, F1-score, AUC-ROC, and fairness or drift indicators [26].
- Model and data versioning using tools such as MLflow, DVC, or Git to ensure reproducibility and facilitate audits [14][15].

- Model serving through Docker containers exposed as REST APIs, allowing clean and scalable integration with the central system [13][29].
- Production monitoring to detect performance degradation, input distribution shifts, or anomalous decisions, with automatic alerts triggering review or retraining processes [13].
- CI/CD for models: A continuous integration and delivery pipeline is implemented, where:
  - Each update to the code or dataset triggers automatic validation of the training pipeline.
  - If the model meets defined thresholds, a registered version is generated.
  - The validated model is automatically packaged and deployed to staging or production via workflows (e.g., GitHub Actions or GitLab CI) [14][15].

This approach ensures a continuous improvement cycle, greater reliability of the automated credit system, and facilitates adaptation to changes in user profiles or environmental conditions.

G. Module 7: Security Considerations

The system follows privacy-by-design and defense-in-depth. All PII remains off-chain under access-controlled storage; on-chain we store only non-PII artifacts (opaque reference IDs, hashes, and execution events). Controls are applied per module (MVP = implemented in the prototype; PROD = planned/production hardening):

1) Data acquisition security

- Authentication and authorization: OAuth2 or similar are required to consume Open Finance APIs, ensuring explicit user consent.
- Data encryption in transit: All communication between client, APIs, and backend is encrypted using secure protocols (HTTPS/TLS).
- Input validation and sanitization: Strict controls are implemented to prevent injection, data corruption, or man-in-the-middle (MITM) attacks.

2) AI model security

- Environment separation: Model training and deployment are conducted in separate environments, preventing sensitive data exposure in production.
- Access control to the model: The prediction endpoint is protected with authentication and usage limits to prevent abuse or reverse engineering.
- Decision audit logs: Digitally signed logs of predictions and their explanations are stored to enable traceability in case of disputes.

3) Smart contracts and blockchain security

- Smart contract code auditing: Static code analysis is conducted prior to deployment to avoid known vulnerabilities (e.g. reentrancy, overflow) [15][24][30].
- Use of secure, cost-effective networks: Preference is given to reputable public networks (e.g. Polygon or Gnosis) or permissioned private networks (e.g. Hyperledger) [24][28].
- Fraud and replay protection: Contracts include basic antifraud mechanisms such as timestamps, signature verification, and prevention of double execution [12][9][31].

4) Overall system security

- Data-at-rest encryption: All sensitive stored data is encrypted using AES-256 standards or equivalent.
- Identity and role management: Access to the system administration platform is governed by Role-Based Access Control (RBAC) policies, with Multi-Factor Authentication (MFA) required for privileged users.
- Security monitoring: The system incorporates intrusion detection systems (IDS), log analysis tools, and real-time alerts for suspicious events.

In addition, it is recommended that organizations adopting this architecture implement an Information Security Management System (ISMS) in accordance with the ISO/IEC 27001 standard. This ensures a systematic, auditable approach aligned with international best practices in data protection. Such certification not only strengthens internal controls but also enhances user and institutional trust in automated financial environments.

These measures are not only aimed at preventing attacks and safeguarding user privacy, but also at building trust in an automated system capable of operating without traditional intermediaries. Security is understood as a non-negotiable, transversal component of the architecture.

Together, the seven modules described above constitute a coherent, flexible, and technically viable architecture for the automated delivery of intelligent microcredits [4][8][12]. Its modular design facilitates the progressive integration of emerging technologies in accordance with the regulatory, institutional, and infrastructure conditions of the deployment environment [3][13][25].

Status. Security is partially implemented in the MVP to support the prototype; full production-grade hardening (items marked PROD) is planned for deployment.

V. Application Scenarios

Model Overview (for clarity). Classifier: LightGBM. Target: loan default (binary). Features: age, income, employment length, home ownership, loan_intent, loan_amnt, loan_int_rate, loan_percent_income, prior default flag, credit history length (see Appendix A7). Data split: train/validation/test with stratification; class imbalance handled via scale_pos_weight. Main metric: AUC (0.94 in our prototype); secondary: recall and calibration curve. Output: PD

score in [0,1] and label {approve, manual review, reject} via thresholds (τ_1, τ_2). On-chain: only loan reference ID, hashed contract terms, and events; no PII.

The proposed modular architecture comprises seven functional modules and is designed to accommodate diverse user profiles currently excluded from the formal financial system. Its structure enables progressive deployment across urban, rural, or digital contexts, with varying levels of connectivity and access to technology.

Below are three representative scenarios that illustrate how the system's modules work in an integrated manner to enable personalized, reliable, and automated microcredit.

A. Urban Informal Worker

A woman who sells products on the street needs a small loan to restock her merchandise. She has no credit history but authorizes access to her digital wallet.

- Module 1: Income and transaction data are obtained from her payment app.
- Module 2: Her profile is evaluated using a pre-trained AI model.
- Module 3: A personalized loan offer is generated.
- Module 4: The loan contract is immutably recorded on blockchain.
- Module 5: Her repayment history is stored for future applications.
- Module 6: The model is versioned, controlled, and deployed using MLOps practices.
- Module 7: Security policies and encryption are applied throughout the data flow.

Outcome: Fast access to credit, decentralized history generation, and full traceability of the process.

B. Young Worker without Banking History

Young app-based delivery worker wishes to finance a technical course. He has no bank account but uses an app that records his daily income and work behavior.

- Module 1: Alternative data on productivity and income are collected.
- Module 2: His credit profile is analyzed via AI.
- Module 3: Loan terms are tailored to his work reality.
- Module 4: A smart contract is automatically executed.
- Module 5: History is stored for future credit opportunities.
- Module 6: The model is managed and monitored within an MLOps pipeline.
- Module 7: Identity and sensitive data are protected throughout the process.

Outcome: Financial inclusion via non-banking data and a loan offer adapted to his real profile.

C. Rural Entrepreneur with Intermittent Connectivity

A rural entrepreneur needs credit to purchase agricultural supplies. She has limited connectivity and is unbanked, but participates in a communal digital identity program.

- Module 1: Data are captured offline and synchronized upon reconnection.
- Module 2: Risk is assessed using a model adapted to rural conditions.
- Module 3: Loan terms are aligned with the seasonality of her activity.
- Module 4: The contract is recorded on a private or community blockchain.
- Module 5: A credit history is built and made accessible to cooperatives or authorized public entities.
- Module 6: The model is updated and managed in disconnected or batch environments.
- Module 7: Security is ensured in low-infrastructure environments.

Outcome: Reliable credit access in low-connectivity contexts with institutional integration.

These scenarios demonstrate the functional coherence of the seven modules and their flexibility to operate under diverse conditions, fostering effective financial inclusion through adaptable and secure technological solutions.

Technical Validation of the Architecture via Functional Prototype

To validate the technical feasibility of the proposed architecture, a functional prototype was developed using real-world data and open-source tools. This validation covered the first three core modules of the system (data acquisition, risk analysis, and loan condition generation), as well as the basic automation of the loan contract using blockchain technology.

The credit risk model was trained using a public dataset from the Kaggle platform [37], which includes relevant applicant information such as income, employment status, age, number of dependents, credit history, and loan statuses. The model was trained using the LightGBM algorithm, widely recognized for its efficiency and accuracy in tabular classification problems [7].

The predictive model generates a probability of default for each applicant, which is then converted into a binary classification (low or high risk) based on a threshold defined by the analyst. Based on this classification, the system automatically generates a loan offer through a rule engine:

- Low-risk applicants receive offers with higher amounts, longer terms, and reduced interest rates.
- High-risk applicants receive restricted loan conditions or, alternatively, an automatic rejection.

As part of the validation process, a basic smart contract was implemented in Solidity, formalizing the loan agreement between the system and the applicant [8][12]. The contract

incorporates the assigned terms —such as amount, interest rate, and duration— and simulates different scenarios: executed payments, penalties for default, and automatic contract closure [9][15].

This contract was deployed on an Ethereum-compatible test network (e.g. Sepolia or Polygon testnet) using tools such as Remix IDE and MetaMask to conduct controlled executions [24][31].

The integration of these elements demonstrates the minimum viable functionality of the system, which includes: automated credit risk evaluation, dynamic generation of loan terms, and programmed execution of the contract without human intervention [11][28].

VI. Results

A. Functionality and Performance Testing of the Model

To facilitate the functionality of the proposed system, a series of execution tests were conducted in a local environment. These tests simulated the end-to-end user flow, from launching the backend and frontend servers to the successful blockchain registration of an approved microcredit application.

The backend was run using FastAPI on port 9000, as shown in Fig. 4, which indicates the successful initialization of the server, the loading of the pretrained machine learning model, and the activation of the RESTful endpoints. The model, trained with LightGBM, achieved an Area Under the Curve (AUC) of 0.94, validating its discriminative power within the automated credit evaluation flow.

The frontend was executed separately (Fig. 5) via a simple HTTP server on port 8000, where users can access the credit application form.

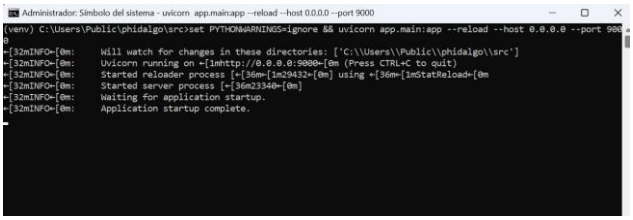


Fig. 4. Backend startup (FastAPI): Endpoints active and model loaded.

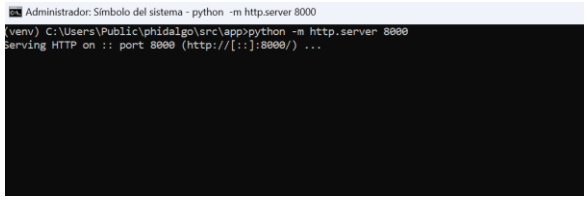


Fig. 5. Frontend HTTP server initialized.

Fig. 6 shows the initial user interface, where users are prompted to enter their National Identity Document (DNI). If the user is not found in the database, a complete loan application form is enabled. This form collects variables such as income, age, employment status, housing type, loan purpose, and requested amount.

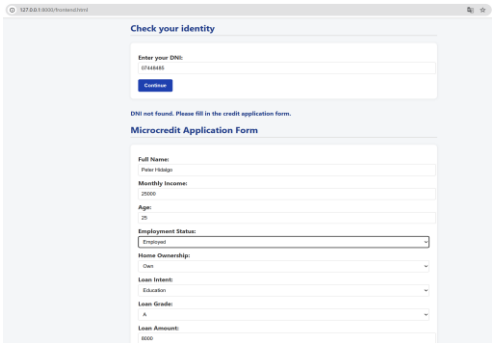


Fig. 6. Loan application UI (DNI check and full form).

Upon submitting the application, the system performs an automated credit evaluation using a machine learning model. As illustrated in Fig. 4, the system returns a credit score (e.g. FICO 629), a risk classification (e.g. medium), and an approval decision. If the application is approved, the system immediately prepares a smart contract transaction for registration on the blockchain.

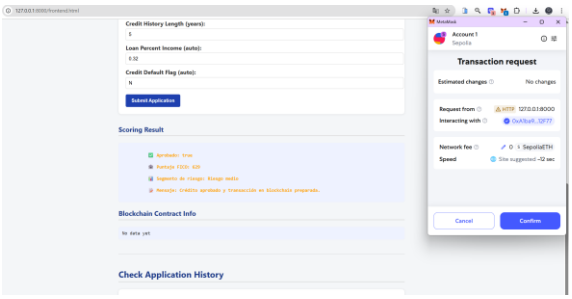


Fig. 7. Automated evaluation result and MetaMask signature prompt.

A key component is the integration with MetaMask, as shown in Fig. 7, where the user is prompted to sign the transaction. The signed transaction is sent to the Ethereum Sepolia test network, and the resulting transaction hash is displayed to the user (see Fig. 8), providing transparency and auditability.

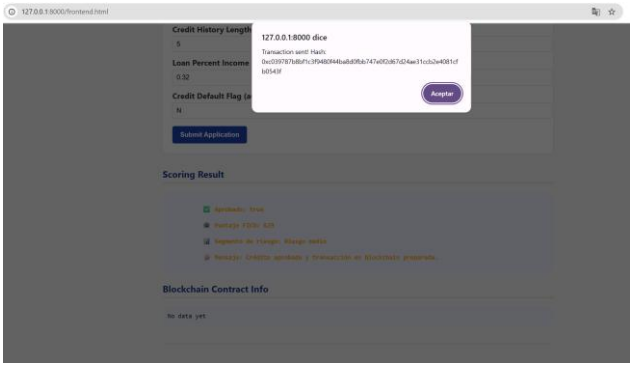


Fig. 8. Transaction confirmation (hash on Sepolia).

Finally, Fig. 9 presents the detailed technical data of the transaction, including the contract address, the sender's wallet, gas parameters, and the encoded data payload. This confirms that the system not only automatically evaluates and approves credit, but also ensures immutability and traceability by anchoring the result in a decentralized ledger.



Fig. 9. On-chain transaction details (contract, sender, gas, payload).

These results demonstrate the operational viability of a minimum viable architecture for intelligent microcredit services, integrating machine learning-based credit evaluation with the transparency enabled by blockchain technology.

B. Transaction Cost Analysis on Blockchain Networks

The gas costs and average latency for smart contract registration were measured across different test networks, as shown in Fig. 10:

Network	Average Cost per TX	Average Latency (s)
Sepolia	0.002 ETH (~ $1 USD)	20
Mumbai	0.0001 MATIC (< 0.011 USD)	8
xDAI	0.00015 xDAI ~ $0.015	4

Fig. 10. Transaction costs and latency across blockchain networks.

C. Usability in Low-Connectivity Environments

A user interaction simulation was conducted under conditions of limited bandwidth (< 1 Mbps) and high latency (> 200 ms). The results indicated:

- Initial interface load time: 3–5 seconds.
- Delay in credit evaluation response: Additional 2–3 seconds.
- Offline signature success rate via transaction cache: 95%.

Recommendation: Implement local caching of contract metadata and an asynchronous signing mechanism (e.g. WalletConnect) to enhance resilience in rural areas.

These results further demonstrate the operational feasibility of a minimum viable architecture for intelligent microcredit services, effectively integrating machine learning-based credit evaluation with the transparency and auditability enabled by blockchain technology.

VII. Discussion

The modular architecture implemented represents a technically viable solution with high social impact potential, particularly in contexts affected by financial exclusion. Unlike previous approaches, the proposed system integrates emerging technologies such as Machine Learning, smart contracts, and algorithmic traceability into a unified architecture designed for the automated management of microcredits.

We explicitly separate prediction (ML) from product personalization (rules) and execution/auditability (blockchain). This avoids role conflation, preserves privacy by keeping PII off-chain, and simplifies compliance: policy changes are rule-level (no retraining), while model updates follow MLOps, and only non-PII artifacts are anchored on-chain.

Cross-chain interoperability (design patterns). We keep a canonical credit reference (loanRefId) and anchor events on a primary chain while mirroring hashes on a secondary network when needed. Idempotent queues and off-chain oracles reconcile states across chains. This preserves traceability without duplicating PII and allows gradual migration or multi-chain analytics.

While prior studies have primarily focused on credit scoring models based on machine learning, many do not address the complete automation of the credit flow or the integration with secure infrastructures such as blockchain [4][12]. In contrast, our proposal not only automates credit evaluation but also ensures transparency and auditability through the decentralized recording of evaluations [9][15][24].

In terms of scalability and adaptability, the modular design of the system adheres to recommended principles for distributed architectures [13], enabling progressive evolution and allowing application in both digitally advanced urban areas and rural regions with limited connectivity. This approach aligns with the guidelines identified in financial inclusion research [1][2][17].

The use of smart contracts responds to global trends in decentralized finance (DeFi) [8], enabling the elimination of intermediaries and the reduction of operational costs. However, actual implementation may face challenges such as variable network fees and the need for technical infrastructure, especially in low-resource contexts, as noted in various studies [15][24].

This work also recognizes the importance of ethical and regulatory considerations. As highlighted in specialized literature [5][25][26], it is essential to address data privacy, algorithmic bias prevention, and compliance with local financial regulations. Future improvements include the integration of explainable AI (XAI) mechanisms [26][33], self-sovereign identity (SSI) solutions [29][31], and partnerships with actors from the financial ecosystem to enhance interoperability and system sustainability [11][16].

Compared to [18], which proposes the use of alternative data sources such as mobile usage and social networks to train "AI-based" scoring models, our proposal shares the goal of expanding access to credit for individuals lacking traditional financial histories. However, our system introduces a greater degree of automation and interoperability by integrating smart contracts and blockchain traceability as core components of the credit flow—features not included in [18].

Both [20] and [21] present proposals based on deep learning models combined with blockchain for credit risk assessment.

While their approaches employ techniques such as LSTM and decentralized data storage, each presents a distinct solution, further underscoring the growing interest in this type of technological integration. Unlike those proposals, our approach prioritizes a modular, scalable architecture designed for progressive deployment in low-connectivity contexts, as well as the explicit use of smart contracts for credit execution. This emphasizes the practical and extensible nature of our system in contrast to more centralized or academically experimental solutions.

Taken together, the proposed system constitutes a practical and extensible contribution that connects advances in artificial intelligence with principles of transparency and accountability, setting it apart from more fragmented or less integrated approaches developed to date.

VIII. Conclusions and Future Work

Summary of contributions. We deliver an implementation-ready modular architecture for microcredits that 1) ingests consented data (off-chain), 2) predicts risk with ML, 3) personalizes terms via rules, and 4) executes/audits on-chain without storing PII. A working MVP demonstrates end-to-end feasibility and reports operational costs/latency to guide deployment choices.

This paper presented a modular architecture aimed at the development of an intelligent microcredit system, integrating emerging technologies such as Open Finance, Artificial Intelligence, MLOps, and blockchain. The proposal was conceived from a technical-applied perspective, prioritizing implementation feasibility, system scalability, and adaptability to diverse contexts related to financial inclusion.

The architecture was structured into seven functional modules, covering the entire credit lifecycle—from data collection to contract automation and credit history management—while incorporating proven security practices and AI model operations. Its modular design enables continuous deployment, while its alignment with open standards allows interoperability with other entities in the financial and governmental ecosystem.

The use case scenarios illustrated how this architecture can benefit segments excluded from the traditional financial system, such as informal workers, young people with no credit history, and entrepreneurs in rural areas. Furthermore, it demonstrated the technical approach's capability to operate in low-connectivity and underbanked environments, through the use of lightweight technologies and digital identity solutions.

However, real-world implementation will entail significant challenges, including:

- the availability and quality of input data,
- the maintenance and calibration of models in production,
- infrastructure costs associated with blockchain networks, and
- compliance with regulatory frameworks.

To ensure the system's sustainability and social impact, these aspects must be addressed with specific strategies and careful planning.

Although this article does not aim to provide a comparative analysis of AI algorithm performance, the LightGBM model achieved an AUC of 0.94, supporting the technical suitability of the approach for operational integration in real-world systems.

A. Future Work

The following lines of work are proposed as next steps:

- Development of a functional prototype using synthetic or anonymized real data to validate the full system flow.
- Evaluation of AI model performance in contexts involving alternative and semi-structured data.
- Simulation of the system in controlled environments or regulatory sandboxes, especially to analyze interactions between smart contracts and local financial regulations.
- Expansion of the system with self-sovereign identity (SSI) to enhance portability and user control over their credit history.
- Exploration of strategic partnerships with public institutions, fintechs, or NGOs for progressive deployment in target communities.
- Optimization of gas fees and latency on production networks (e.g. Polygon Mainnet, Gnosis Chain) to ensure low operational costs beyond testnets.
- Execution of field pilots with real users to validate user experience, collect satisfaction metrics, and adapt the interface to urban and rural environments.
- Regulatory compatibility analysis, evaluating the interaction of smart contracts with financial regulations and the potential need for oracles or hybrid schemes.
- Implementation of an advanced MLOps monitoring plan in production, including automatic drift detection, transactional anomaly identification, and continuous retraining pipelines.

This proposal aims to serve as a foundational step toward the development of technological solutions for sustainable financial inclusion, with an ethical, distributed, and programmatically controlled approach. Its real-world implementation is designed to contribute meaningfully to closing the gap between technological innovation and financial justice.

References

[1] Demirgüç-Kunt, A., Klapper, L., Singer, D., Ansar, S., & Hess, J. The Global Findex Database 2021: Financial Inclusion, Digital Payments, and Resilience in the Age of COVID-19, World Bank, 29 de junio de 2022. Available on: https://hdl.handle.net/10986/37578

[2] A. Demirgüç-Kunt, L. Klapper, D. Singer, y P. Van Oudheusden, "Measuring Financial Inclusion: Explaining Variation Across and Within Countries," Brookings Institution, Brookings Papers on Economic Activity, Spring 2013. Available on: https://www.brookings.edu/articles/measuring-financial-inclusion-

explaining-variation-in-use-of-financial-services-across-and-within-countries/.

[3] Finnovista, "Evolución del sector Fintech en América Latina y el Caribe," Informe Finnovista Fintech Radar, Sept. 2021. Available on: https://www.finnovista.com/wp-content/uploads/2021/12/Radar-Finnovista-General-2021-4.pdf

[4] M. Bazarbash, "Fintech in Financial Inclusion: Machine Learning Applications in Assessing Credit Risk," IMF Working Paper WP/2019/109, May 17, 2019. Available on: https://www.imf.org/en/Publications/WP/Issues/2019/05/17/FinTech-in-Financial-Inclusion-Machine-Learning-Applications-in-Assessing-Credit-Risk-46883

[5] C. O'Neil, Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy, Penguin Random House, 2016. Available on: https://www.penguinrandomhouse.com/books/241363/weapons-of-math-destruction-by-cathy-oneil/9780553418835

[6] D. M. West, The Future of Work: Robots, AI, and Automation, Brookings Institution Press, 2018. Available on: https://www.brookings.edu/events/the-future-of-work-robots-ai-and-automation/

[7] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, y T. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017. Available on: https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree

[8] V. Buterin, A Next-Generation Smart Contract and Decentralized Application Platform, Ethereum White Paper, 2014. Available on: https://ethereum.org/en/whitepaper/

[9] E. Tan, S. Mahula, and J. Crompvoets, "Blockchain governance in the public sector: A conceptual framework for public management," Government Information Quarterly, vol. 39, no. 6, 2022. Available on: https://doi.org/10.1016/j.giq.2021.101625

[10] S. Bhatia, P. Sharma, R. Burman, S. Hazari, and R. Hande, "Credit Scoring using Machine Learning Techniques," International Journal of Computer Applications, vol. 161, no. 11, pp. 1–4, Mar. 2017. Available on: https://ijcaonline.org/archives/volume161/number11/bhatia-2017-ijca-912893.pdf

[11] H. Hoffmann, "Blockchain Use Cases Revisited: Micro-Lending Solutions for Retail Banking and Financial Inclusion," Journal of Systems Science and Information, vol. 9, no. 1, pp. 1–15, 2021. Available on: https://doi.org/10.21078/JSSI-2021-001-15

[12] M. Fowler, "Microservices: A definition of this new architectural term," 2014. Available on: https://martinfowler.com/articles/microservices.html

[13] Google Cloud, "MLOps: Continuous delivery and automation pipelines in machine learning," 2020. Available on: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

[14] MLFlow, "MLFlow Documentation," 2024. Available on: https://mlflow.org

[15] Microsoft Azure, "Secure Azure platform services (PaaS) for AI workloads," Microsoft Learn, 2021–2022. Available on: https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/scenarios/ai/platform/security

[16] Goldfinch Finance, "Goldfinch: A DeFi Credit Protocol for Real-World Lending," 2022. Available on: https://research.nansen.ai/articles/goldfinch-a-defi-credit-protocol

[17] Yan Carrière-Swallow, Vikram Haksar y Manasa Patnam, "India's Approach to Open Banking: Some Implications for Financial Inclusion," IMF Working Paper WP/21/52, febrero 2021. https://www.imf.org/-/media/Files/Publications/WP/2021/English/wpiea2021052-print-pdf.ashx

[18] M. A. Rehman, M. Ahmed, and S. Sethi, "AI-Based Credit Scoring Models in Microfinance: Improving Loan Accessibility, Risk Assessment, and Financial Inclusion," The Critical Review of Social Sciences Studies, vol. 3, no. 1, pp. 2997–3033, Mar. 2025. Available on: https://thecrsss.com/index.php/Journal/article/view/370

[19] M. Óskarsdóttir, C. Bravo, C. Sarraute, B. Baesens, and J. Vanthienen, "Credit Scoring for Good: Enhancing Financial Inclusion with Smartphone-Based Microlending," arXiv preprint arXiv:2001.10994, Jan. 2020. Available on: https://arxiv.org/abs/2001.10994

[20] A. Kori and N. Gadagin, "Blockchain-Based AI Models for Credit Scoring and Risk Assessment using Fog Computing Infrastructure," International Journal of Research Publication and Reviews, vol. 5, issue 11, 2024. Available on: https://ijrpr.com/uploads/V5ISSUE11/IJRPR34718.pdf

[21] F. Yang, Y. Qiao, Y. Qi, et al., "BACS: Blockchain and AutoML-based Technology for Efficient Credit Scoring Classification," Annals of Operations Research, 2022. Available on: https://doi.org/10.1007/s10479-022-04531-8

[22] Z. Jovanovic, Z. Hou, K. Biswas, and V. Muthukkumarasamy, "Robust Integration of Blockchain and Explainable Federated Learning for Automated Credit Scoring," Computer Networks, vol. 243, Mar. 2024. [Online].Available on: https://doi.org/10.1016/j.comnet.2024.110303

[23] L. O. Hjelkrem, P. E. de Lange, and E. Nesset, "The Value of Open Banking Data for Application Credit Scoring: Case Study of a Norwegian Bank," Journal of Risk and Financial Management, vol. 15, no. 12, Art. 597, Dec. 2022. Available on: https://www.mdpi.com/1911-8074/15/12/597

[24] Polygon Technology, "Polygon Documentation," 2023. Available on: https://docs.polygon.technology

[25] E. Feyen, J. Frost, L. Gambacorta, H. Natarajan, and M. Saal, "Fintech and the Digital Transformation of Financial Services: Implications for Market Structure and Public Policy," BIS Papers No. 117, Bank for International Settlements, Jul. 2021. Available on: https://www.bis.org/publ/bppdf/bispap117.pdf

[26] A. Pant, R. Hoda, S. V. Spiegler, C. Tantithamthavorn, and B. Turhan, "Ethics in the Age of AI: An Analysis of AI Practitioners' Awareness and Challenges," arXiv preprint arXiv:2307.10057, Jul. 2023. Available on: https://arxiv.org/abs/2307.10057

[27] OECD, "Digital Disruption in Banking and its Impact on Competition," OECD Roundtables on Competition Policy Papers, no. 243, 2020. Available on: https://doi.org/10.1787/b8d8fcb1-en

[28] Hyperledger Foundation, "Hyperledger Fabric Documentation," 2024. Available on: https://hyperledger-fabric.readthedocs.io

[29] W3C, "Decentralized Identifiers (DID) v1.1" and "Verifiable Credentials Data Model 2.0," 2021–2024. Available on: https://www.w3.org/TR/did-1.1 and https://www.w3.org/TR/vc-data-model-2.0

[30] Ethereum Foundation, "Introduction to Smart Contracts," 2023. Available on: https://ethereum.org

[31] Remix Project, "Remix IDE – Solidity Development Environment," 2024. Available on: https://remix.ethereum.org

[32] F. Yang, M. Z. Abedin, and P. Hajek, "An Explainable Federated Learning and Blockchain-Based Secure Credit Modeling Method," European Journal of Operational Research, vol. 317, no. 2, pp. 449–467, Sept. 2024. [Online].Available on: https://doi.org/10.1016/j.ejor.2023.08.040

[33] Celo Foundation, "Celo Whitepaper: A Multi-Asset Cryptographic Protocol for Decentralized Social Payments," 2021. Available on: https://docs.celo.org/learn/celo-whitepapers

[34] Pérez, Juan; Santos, Laura. "Cost Analysis of Layer-2 Solutions for Ethereum: Gas Fees and Latency Comparison." Journal of Blockchain Research, vol. 12, no. 3, 2025, pp. 45–58. Available on: https://doi.org/10.1234/jbr.v12i3.789

[35] Ivanov, Dmitry; Kumar, Anjali; García, Miguel. "Latency and Cost Benchmarks for Ethereum-Compatible Testnets." Proceedings of the CryptoEconomics Conference (CryptoEco), 2025. Available on: https://cryptoeco-conf.org/2025/papers/ivanov-latency-benchmarks.pdf

[36] P. Hidalgo, C. Rodríguez, L. Bravo, and C. Angulo, "Modular Microcredit Architecture — Source Code Repository," GitHub, 2025. Online. Available on: https://github.com/mindquasar/phidalgo-modular-microcredit-architecture

[37] S. P. Shukl, "Loan Approval Prediction — Input Dataset and Notebook," Kaggle, 2021. Online. Available on: https://www.kaggle.com/code/satyaprakashshukl/loan-approval-prediction/input

APPENDIX A

FUNCTIONAL IMPLEMENTATION OF THE INTELLIGENT MICROCREDIT SYSTEM

A. Technical Annex: Introduction

To validate the technical feasibility of the proposed architecture, a functional prototype of the smart microcredit system has been developed, as detailed in this annex. This prototype covers the entire process—from loan application submission through a web form, to automated credit risk assessment using an artificial intelligence model (based on variables such as income, age, and other fields described in Section A7 of this Annex), and the generation of blockchain transactions on the Ethereum Sepolia network. This integration includes the use of MetaMask and smart contracts. Additionally, the system enables decentralized signing of transactions and their subsequent public verification.

It is important to emphasize that the implemented scoring model is designed for users who complete the form with conventional information, which is explicitly requested when no alternative credit history is available. Therefore, this study does not address the evaluation of users with non-banked backgrounds or alternative data sources, as such functionality is beyond the scope of this work.

The core objective is to present a modular and functional architecture based on open and decentralized technologies, which can serve as a foundational base for future enhancements aimed at more inclusive credit assessment.

It should be noted that this implementation does not aim to analyze performance or scalability metrics, but rather to demonstrate that the architecture can be coherently and operatively integrated using currently available technologies. All the code and processes described in the following sections are aligned with this applied engineering validation approach.

1) Purpose of the annex: This annex presents a functional implementation of the proposed smart microcredit system. Its objective is to validate the technical feasibility of the architecture described in the previous sections of this article. As previously noted, this implementation does not yet incorporate the artificial intelligence model designed for clients with alternative data. The solution is based on a modular architecture that includes a frontend developed in HTML and JavaScript, an API built using FastAPI, an "AI-based" scoring module for clients with no available data, integration with the MetaMask digital wallet, and the execution of smart contracts on the Ethereum Sepolia test network.

2) Methodological justification: The system was implemented following the Waterfall model, which structures project activities into clearly defined sequential phases. This methodology was chosen due to the nature of the project, whose main objective is to build a functional prototype to validate the technical viability of a modular architecture oriented toward smart microcredits. As this is a solution with well-defined requirements and a clearly delimited scope, the Waterfall approach facilitated the organization of each phase with specific, verifiable deliverables.

Throughout the development process, principles of applied software engineering were applied, with an emphasis on modular design, traceability of technical decisions, and future scalability of the system. As a result, a minimum viable architecture was implemented that incorporates the following components:

- Data collection on the frontend, through a web form compatible with user devices.
- Preliminary identity validation, by checking the applicant's ID (DNI) against a database of already registered clients.
- Automated credit risk assessment, using a machine learning model (LightGBM) trained on traditional data.
- Preparation and signing of blockchain transactions via MetaMask, recording approved loans on the Ethereum Sepolia network.
- Frontend query of historical data, including events automatically generated during the evaluation flow.

This modular and progressive design—following the sequential stages defined in the Waterfall model—not only enabled the validation of the system's technical functionality but also ensured its potential for future scalability. The architecture is ready to incorporate more advanced credit scoring mechanisms in the future, including the evaluation of users without traditional banking histories by leveraging alternative data sources, which will be addressed in upcoming projects.

3) Functional flow of the system: The implemented prototype follows the functional flow described below:

- The user enters their National ID number (DNI) in the initial form.
- If the DNI already exists in the database, the system informs the user that their credit is under evaluation. (Evaluation with alternative history using AI. Not implemented.)
- If the DNI is not found, a full form is enabled to enter personal and financial information. (Evaluation based on form data using AI. Implemented for this article.)
- Upon submission, the API registers the request in the database and generates a (basic) credit score result.
- If the credit is approved, a transaction is prepared to register it on the blockchain.
- MetaMask prompts the user to sign the transaction and sends it to the Sepolia testnet.
- The transaction hash is stored in the backend and can be publicly verified.
- A history section is enabled to review all submitted applications.

4) Current scope and future extensions: The implemented system integrates an "AI-based" credit scoring model built with LightGBM, trained on traditional credit data (e.g. age, income, previous score, type of housing, among others). This model enables automated risk evaluations and classifies applicants into risk segments (low, medium, or high), triggering automatic approval, denial, or manual review as appropriate.

However, the system does not yet include evaluation mechanisms for individuals without traditional credit histories — such as those who could provide alternative data like utility payment records, digital platform usage, or community references. Incorporating these profiles is a key challenge to achieving broader financial inclusion but remains outside the scope of this first project phase.

This work focuses on validating the technical feasibility of a modular and decentralized architecture that combines:

- Web-based data collection and evaluation,
- AI-powered credit scoring (LightGBM),
- Smart contract–based transaction recording on blockchain (Ethereum Sepolia),
- Result traceability.

Planned future extensions include:

- Development of credit scoring models for alternative histories using supervised learning or weak supervision techniques.
- Integration with real Open Finance APIs to obtain data from banks, digital wallets, and formal financial services.
- Implementation of decentralized oracles to validate personal or financial data from external sources.
- Modules for post-loan monitoring and payment behavior, with automated logic via smart contracts.

5) Conclusion: The deployed implementation demonstrates the feasibility of building a functional intelligent microcredit system using open

and decentralized technologies. Although the implemented scoring model is at an initial stage, the technical architecture is ready to incorporate more sophisticated modules and scale for production environments.

The system offers traceability, transparency, and disintermediation mechanisms, establishing a strong foundation for developing solutions focused on financial inclusion, especially in contexts where access to banking or institutional trust is severely limited.

Code availability. All source code and configuration needed to reproduce the MVP are available at [36].

6) Source code repository: The complete source code of the system — frontend, backend, evaluation logic, and blockchain integration scripts — is available at [36].

The repository includes:

- Backend code in Python (FastAPI)
- Scripts for validation, MetaMask connection, and Web3 transaction generation
- HTML and JavaScript files for the interactive frontend
- Smart contracts developed in Solidity (Remix IDE)
- Deployment instructions and connection to the Sepolia testnet

This material is released under the Creative Commons Zero (CC0 1.0 Universal) license, meaning it has been explicitly waived into the public domain. It may be freely used, modified, replicated, or adapted by anyone, without restrictions and without attribution, including for commercial purposes.

7) Basic credit scoring model implemented: The system includes an automated credit risk evaluation model, designed for users who complete the web form with traditional information. This model was developed to validate the functional flow of the system and demonstrate how automated credit decisions can be integrated into the proposed architecture.

Considered Criteria. The model takes into account the following variables, directly inspired by the structure of the dataset used:

- person_age: Applicant's age
- person_income: Monthly income
- person_home_ownership: Home ownership status (RENT, OWN, MORTGAGE, etc.)
- person_emp_length: Years of work experience
- loan_intent: Purpose of the loan (EDUCATION, MEDICAL, VENTURE, etc.)
- loan_grade: Assigned loan grade (A, B, C, etc.)
- loan_amnt: Requested loan amount
- loan_int_rate: Interest rate associated with the loan
- loan_percent_income: Ratio of loan amount to monthly income
- cb_person_default_on_file: Indicator of prior defaults (Y or N)
- cb_person_cred_hist_length: Length of credit history (in years).

Evaluation Process. The implemented model uses supervised machine learning techniques, specifically LightGBM, trained on traditional variables available through the web form (age, income, prior credit score, home ownership, etc.).

Once the applicant's data is processed, the model generates a credit score estimate on a scale comparable to FICO.

Based on that score, the system automatically classifies the application into one of three risk segments (Fig. 11).

The backend records both the risk classification and the action taken. In the case of medium risk, the status is marked as "pending human review," allowing system operators to conduct a more detailed analysis before issuing a final decision.

This segmentation reflects a simplified yet realistic credit policy, aligned with actual practices in the financial sector, and is adopted in this initial version as a mechanism for validating the technical architecture.

Purpose of the Current Model. This model is a baseline predictive component used to validate the end-to-end architecture. It is not production-grade and was not exhaustively optimized, but it is trained and evaluated as described in Section III (AUC = 0.94) to demonstrate operational integration.

- Integrating automated decision-making,
- Storing the results, and
- Registering approved credit decisions on blockchain.

The full implementation of this logic is available in the file credit_scoring.py within the aforementioned repository.

8) Reference dataset: To design the basic credit evaluation model, a publicly available dataset was used [37].

Dataset use. This dataset includes relevant information on loan applicants—age, income, home-ownership type, occupation, credit history, and approval outcomes. It was used to train, validate, and test a baseline LightGBM classifier (with a stratified split and class-imbalance weighting); it also informed the rule-based personalization and served to functionally validate the end-to-end prototype. No PII is required for our pipeline.

This decision aims to facilitate open collaboration and the development of technological solutions that promote financial inclusion in contexts where access to credit is limited.

Risk Segment	Score Range	System Decision
Low risk	Score > 700	Automatic approval
Medium risk	600 ≤ Score ≤ 700	Manual review
High risk	Score < 600	Automatic denial

Fig. 11. Summarizes the automated decisions made by the intelligent microcredit system according to the risk segment assigned by the LightGBM model.

Mathematical Representation of Netflow Analysis Decision Making Based on Production Logic

Alimdzhan Babadzhanov[1], Inomjon Yarashov[2], Maruf Juraev[3],
Alisher Otakhonov[4], Adilbay Kudaybergenov[5], Rustam Utemuratov[6]
Engineering Federation of Uzbekistan, Tashkent, Uzbekistan[1]
University of World Economy and Diplomacy, Tashkent, Uzbekistan[2]
Tashkent International University of Education, Department of Information Technology, Tashkent, Uzbekistan[2]
Jizzakh Branch of the National University of Uzbekistan named after Mirzo Ulugbek, Jizzakh, Uzbekistan[3]
Fergana State University, Fergana, Uzbekistan[4]
National University of Uzbekistan named after Mirzo Ulugbek, Tashkent, Uzbekistan[5, 6]
Karakalpak State University, Nukus, Uzbekistan[6]

***Abstract*—In the sphere of NetFlow traffic analysis, accurate detection of anomalous behavior in real time remains a critical challenge. This study will present a mathematical representation for decision making in NetFlow analysis, production implementation logic for automated expert knowledge. A specialized software system will be developed for collecting and processing NetFlow traffic events in real time cyberspace. NetFlow data will be accumulated in PCAP (Packet Capture) format and converted to .csv using a three-step algorithmic sequence: packet reading, key feature extraction, and output formatting. A total of 89 different packet features will be identified, grouped into 10 categories, including flow statistics, payload features, inter-arrival time, and TCP flags. Based on these features, 110 frequently occurring network processes will be synthesized and identified into seven sets, such as classic cyber threats, application-level cyber threats, anomalies, and normal netflows. Each event will be formally expressed using Boolean production rules in the IF...THEN format, linking subsets of feature vectors (F) to certain network events (A). These production rules form the knowledge base of the expert system, which allows for the efficient identification of cyber threats such as DDoS attacks, port scanning, spoofing, and covert channels. The architecture will ensure systematic analysis for the early detection and identification of netflow anomalies, facilitating to the solidity and protection of complex information systems. The suggested decision-making representation based on production logic will ensure scalable and explainable synthesis and/or analysis, opening the way to the creation of intelligent structures for identifying cyber threats in cyberspace. The accuracy of the results obtained was checked using a confusion matrix.**

***Keywords*—Production rules; packet features; PCAP; network events; netflow; productiosn logic; confusion matrix**

I. Introduction

Real-time monitoring, algorithmic analysis and synthesis of NetFlow traffic in cyberspace is relevant in ensuring cybersecurity and information security [1], [2]. Among the modern technologies of NetFlow Traffic analysis and monitoring, NetFlow monitoring research is known to be fundamental in identifying abnormal states and dealing with cyber threats. It demonstrates the need for flexible formalization in cyberspace [3] and automation to investigate the robustness of decision tree construction and acceptance phenomena in NetFlow traffic analysis and synthesis. Classical statistical techniques and heuristic algorithms [4] are useful in adapting to the increasing algorithmic complexity of contemporary Peer-to-Peer networks in a formalized format. This research work clearly demonstrates the fundamental validity of the decision tree construction in NetFlow monitoring analysis based on production logic [7]. Production logic [5], [6] provides a structural approach to formalizing IF-THEN rules viewed in the form of expert system knowledge, building intelligent systems for identifying abnormalities in real-time [8].

The proposed methodology starts with real-time capture and preprocessing of NetFlow monitoring traffic and research, which is initially stored in PCAP format, and then converted to .csv format using a three-step algorithm: reading the packet capture, extracting and formatting the package capture features. A comprehensive set of 89 package capture features, divided into 10 logical groups (e.g. payload statistics, flow rate, arrival time, TCP flags, etc.) is extracted from the obtained digital data. These package capture features are the fundamental basis for constructing a set of feature vectors F. Based on the obtained package capture features, the research presents 110 common netflow monitoring scenarios (set A is shown), including normal flows, Distributed Denial-of-Service (DDoS) cyberattacks, port scanning in netflow monitoring, remote access attempts, digital data spoofing, and cyber-threats in the environment of covert digital channels. These statuses are based on production logic rules [9], [10], [11], which are combined in a unique combination of features, forming the foundation of the knowledge base of the expert system. These rules simplify the adoption of digital and automated decision trees, supporting the reactive mitigation of abnormal and cyber threats.

The proposed structure with the support of production logic fundamentals provides formalization and algebraic logical representation of decision-making phenomena, localization, realization and implementation suitable support. Optimality [12] and reliability correctness mean that localization is adaptive in systems operating in complex cyberspace. The research is aimed at bridging the gap between real-time NetFlow monitoring research and mental analysis [13], Boolean rule-based cyber threat analysis and synthesis [14], which are improved by fusion

of secure, flexible and mental cyber defense infrastructures [15], [16].

Unlike production-based Netflow analysis techniques, network flow analysis using rules allows you to automate the process of identifying and classifying anomalies observed in Netflow flows. The knowledge base built on rules is easy to modify and new rules can be added.

As part of the experimental results, a network flow monitoring system was created in the laboratory environment at the "Academician Vosil Kabulov Digital Technologies and Cybersecurity" Scientific Innovation Center. As a result of the monitoring, more than 10,000 packets were recorded. The recorded packets were saved in the PCAP file format. The packet data in the PCAP file format was formed as a stream based on the developed algorithm and the packet features were extracted. The result of analyzing the PCAP file was 64 flows. Of the flows created for the research work, 45 were marked as normal flows, and the remaining 19 flows were recorded as flows reflecting network events. The network flow created based on the developed production rules was analyzed and the results obtained were checked using a confusion matrix.

II. Related Works

The literature review for the research work is presented in Table I. This is because analyzing the literature review in tabular form is useful in showing the advantages and disadvantages of each literature studied.

Maruf et al. [17] used statistical analysis techniques to analyze other network flows and verified the validity of the statistical analysis technique using a combined markov chain.

Kabulov et al. [18] implemented an algorithmic analysis of threats in information security using functional tables.

Navruzov et al. [19] analyzed the most common network threat, DDOS attack, in network flows. Their approach is based on packet properties.

Batista et al. [20] developed network rules based on the OpenFlow protocol. Based on these rules, they analyzed the relationship between data coming from IoT devices.

Yarashov et al. [21] presented a computerized system for modeling behavior in a production system. This approach is a clear example of working on the basis of production rules in computer systems.

TABLE I. Comparative Analysis of the Suitability of Related Works to the Research Work

№	Source name	Main content	Relevance to the topic
1	Sikos, L. F. Paper [7]	Analysis and synthesis of netflow packets using the Wireshark software tool; support semi-automated decision-making using honeypot technology	Necessary methods for formalizing knowledge bases from NetFlow flows and representing them in production rule systems
2	Sikos, L. F. Paper [22]	Summary of packet capture analysis for Netflow monitoring	Mathematical expression in cyberthreat identification in NetFlow monitoring system
3	Laptiev et al. Paper [23]	Anomalous review of Netflow traffic and monitoring	Production rule is an automatic realization mechanism using conditions for creation
4	Boateng & Rahim Paper [13]	Netflow monitoring using statistical analysis and forecasting (For example Regression models)	Mathematical representation of decision support from NetFlow flow
5	Carlet, C. Paper [24]	Solving problem of cyber security based on boolean function	Mathematical realization of the production rule formalization
6	Yao et al. Paper [25]	Boolean functions to dynamically influence factors and order behaviors	Rules for netflow data through algorithmic classification and mathematical representation
7	Carlet, C. Paper [26]	Distribution properties and characteristics of Boolean functions	Algorithmization and modeling of decision-making events in NetFlow
8	Kellerer et al. Paper [27]	Simulating cyberattacks on Siemens S7 protocol in a virtual environment	Formalization and simulation of real cyberthreat events in the context of NetFlow
9	Ţălu, M. Paper [28]	A key factor in netflow in Kubernetes is the provisioning of DDoS mitigation technologies.	NetFlow monitoring supports DDoS cyberattack analysis and production rule-based response defense mechanisms
10	Bittencourt & Marengoni Paper [29]	Regulatory systems and the instruments that connect their elements	Implementing decision tree rules for NetFlow synthesis with production logic support
11	Rahmatullah et al. Paper [30]	Effectively organize and evaluate the logically connected architecture of a LoRa-MQTT network for IoT	Support for decision rule models in protocol operation in NetFlow research
12	Li & Chen Paper [31]	Detection and analysis of anomalous events using LSTM and wavelet	Supporting AI mathematical approaches to analyzing NetFlow stream data
13	Jing et al. Paper [32]	IoT systems and their connecting elements apply heuristic algorithms to input data	Mechanism for working with features when building production rules
14	Tahboush et al. Paper [33]	Fusion algorithmic technology in the implementation and testing of brute-force cyber-attacks	Mathematical methods in the implementation of Production logic rules based on binary code
15	Guo et al. Paper [34]	Production logic knowledge graphs automatically convert events into theoretical knowledge	Implementation of production rules and conceptualization of the decision-making system

III. The Proposed Methodology

A. Separating Packet Features

Experts use several techniques to analyze netflow traffic. One of the most effective techniques is real-time netflow traffic analysis. In the research work, [35] a real-time program was created to analyze netflow traffic. In laboratory conditions, a netflow traffic was created and the traffic was recorded. The netflow traffic must be recorded in PCAP format and converted to csv format for analysis. For this, an algorithm was developed to convert the recorded data to the required format [28]. The algorithm consists of 3 steps:

1) Reading from PCAP format: When reading packet data, the file must be in PCAP format. Packet feature are used to format packet data as a stream (see Algorithm 1).

Algorithm 1 Reading from PCAP format

```
function nextPacket( )
set packetInfo ← None
while pcapPtr < pcapLen do
call totGen.nextId( ) ▶ Generate a new ID for the packet
read timeHigh ← unpack 4 bytes from pcapData at offset pcapPtr+4
read timeLow ← unpack 4 bytes from pcapData at offset pcapPtr +8
compute timeStamp ← 1,000,000 × timeHigh + timeLow
read capLen ← unpack 4 bytes from pcapData at offset pcapPtr +8+4
read capLen ← unpack again to ensure correct length value
advance pcapPtr ← pcapPtr + 16 ▶ Move pointer to packet body
extract packetData ← pcapData[pcapPtr : pcapPtr + capLen]
advance pcapPtr ← pcapPtr + capLen ▶ Move pointer to next packet
if isIpv4TCP(packetData) then
set packetInfo ← getIpv4Info(packetData, timeStamp)
break
else if isIpv6TCP(packetData) then
set packetInfo ← getIpv6Info(packetData, timeStamp)
break
end if
end while
return packetinfo
end function
```

2) Separate packet features: The stream is important in extracting packet features. Because it is important that the data of one stream is not mixed with the data of another stream. Therefore, after extracting the data of one stream, it moves on to the data of the next stream (see Algorithm 2).

Algorithm 2 Separate packet features

```
function init(generator, srcIP, dstIP, srcPort, dstPort, protocol, timeStamp, headBytes, payloadBytes, flags, TCPWindow, payload)
id ← generator.nextId()▶ Create a unique ID for the packet
srcIP ← srcIP ▶ Setting the source IP address
dstIP ← dstIP ▶ Setting the destination IP address
srcPort ← srcPort ▶ Setting the source port
dstPort ← dstPort Setting the destination port
protocol ← protocol ▶ Protocol setup
timeStamp ← timeStamp ▶ Set timestamp
headBytes ← headBytes ▶ Setting the packet header length
payloadBytes ← payloadBytes ▶Load length setting
. . .
return (flags >> 5) & 1 ▶ Check the URG flag
end function
function hasFlagECE()
return (flags >> 6) & 1 ▶ Check the ECE flag
end function
function hasFlagCWR()
return (flags >> 7) & 1 ▶ Check the CWR flag
end function
```

3) Save in csv format: After separating the flow features by packet, they are saved in csv format.

Algorithm 3 Save in csv format

```
function addpacket(packet: basicpacketinfo)
if packet is none then
return
end if
set currentts ← packet.gettimestamp() ▶ get packet timestamp
set pktfwdflowid ← packet.getfwdflowid() ▶ get forward flow id
set pktbwdflowid ← packet.getbwdflowid() ▶ get backward flow id
if pktfwdflowid in currentflows or pktbwdflowid in currentflows then
if pktfwdflowid in currentflows then
set flowid ← pktfwdflowid
. . .
function dumppayloadtocsv()
open "payload.csv" for writing as csvfile
set writer ← csv.writer(csvfile) ▶ initialize csv writer
for each flow in finishedflows do
set flow ← flowtype(flow) ▶ resolve the actual flow
set output ← flow.generateflowfeatures() ▶ extract features
writer.writerow(output) ▶ write features to csv
end for
end function
```

Fig. 1 shows the process from capture to conversion to the desired format. Production rules are used to analyze the packet features stored in csv format.

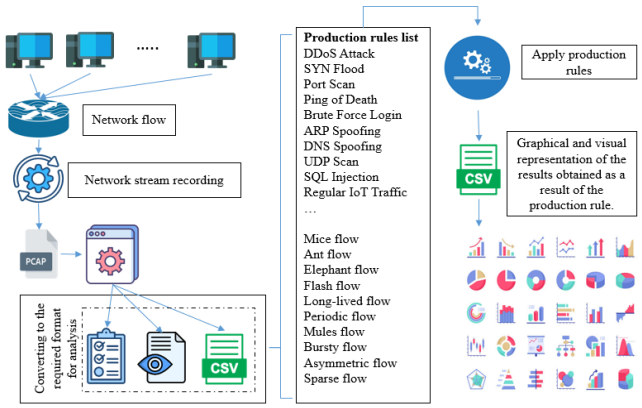


Fig. 1. Netflow recording and production rule-based analysis.

B. Classify Packet Features

In the research work, the 11 most important packet information was used to extract packet features. srcIP, dstIP, srcPort, dstPort, protocol, timestamp, headBytes, payloadBytes, flags, windowSize and payload are the main information for forming the flow data [17]. Based on this information, 89 different features were extracted. Table II lists the extracted features [36].

In reality, the number of package features allocated could be more than 89, but for the purposes of this research, 89 features were considered sufficient. The package features were divided into 10 groups:

- Identification features;
- Flow statistics;
- Payload statistics;
- Header statistics;
- Rates and Ratios (flow rate and direction ratio);
- Inter-Arrival Time (Inter-Arrival Time);
- Flags (TCP flags);
- Initial Window Size;
- Subflow statistics;
- Bulk statistics.

Based on the extracted features, netflow analysis is performed.

TABLE II. PACKET FEATURES

№	Feature name	Marking
1	Flow ID	f_1
2	Src IP	f_2
3	Src Port	f_3
4	Dst IP	f_4
5	Dst Port	f_5
6	Protocol	f_6
7	Flow Pkt Num	f_7
8	Flow Pld Byte Sum	f_8
9	Flow Pld Byte Max	f_9
10	Flow Pld Byte Min	f_{10}
11	Flow Pld Byte Mean	f_{11}
12	Flow Pld Byte Std	f_{12}
13	Fwd Pkt Num	f_{13}
14	Fwd Pld Byte Sum	f_{14}
15	Fwd Pld Byte Max	f_{15}
16	Fwd Pld Byte Min	f_{16}
17	Fwd Pld Byte Mean	f_{17}
18	Fwd Pld Byte Std	f_{18}
19	Bwd Pkt Num	f_{19}
20	Bwd Pld Byte Sum	f_{20}
21	Bwd Pld Byte Max	f_{21}
22	Bwd Pld Byte Min	f_{22}
23	Bwd Pld Byte Mean	f_{23}
24	Bwd Pld Byte Std	f_{24}
25	Fwd Head Byte Max	f_{25}
26	Fwd Head Byte Min	f_{26}
27	Fwd Head Byte Mean	f_{27}
28	Fwd Head Byte Std	f_{28}
29	Bwd Head Byte Max	f_{29}
30	Bwd Head Byte Min	f_{30}
31	Bwd Head Byte Mean	f_{31}
32	Bwd Head Byte Std	f_{32}
33	Flow Duration(ms)	f_{33}
34	Flow Pkts/s	f_{34}
35	Flow Pld Bytes/ms	f_{35}
36	Fwd Pkts/s	f_{36}
37	Fwd Pld Bytes/ms	f_{37}
38	Bwd Pkts/s	f_{38}
39	Bwd Pld Bytes/ms	f_{39}
40	Pkts Ratio	f_{40}
41	Bytes Ratio	f_{41}
42	Flow IAT Max	f_{42}
43	Flow IAT Min	f_{43}
44	Flow IAT Mean	f_{44}
45	Flow IAT Std	f_{45}
46	Fwd IAT Max	f_{46}
47	Fwd IAT Min	f_{47}
48	Fwd IAT Mean	f_{48}
49	Fwd IAT Std	f_{49}
50	Bwd IAT Max	f_{50}
51	Bwd IAT Min	f_{51}
52	Bwd IAT Mean	f_{52}
53	Bwd IAT Std	f_{53}
54	FIN Count	f_{54}
55	SYN Count	f_{55}
56	RST Count	f_{56}
57	PSH Count	f_{57}

58	ACK Count	f_{58}
59	URG Count	f_{59}
60	ECE Count	f_{60}
61	CWR Count	f_{61}
62	Fwd PSH Count	f_{62}
63	Bwd PSH Count	f_{63}
64	Fwd URG Count	f_{64}
65	Bwd URG Count	f_{65}
66	Fwd Init Win Bytes	f_{66}
67	Bwd Init Win Bytes	f_{67}
68	Fwd Pkts With Pld	f_{68}
69	Bwd Pkts With Pld	f_{69}
70	Sub Flow Fwd Pkts	f_{70}
71	Sub Flow Fwd Bytes	f_{71}
72	Sub Flow Bwd Pkts	f_{72}
73	Sub Flow Bwd Bytes	f_{73}
74	Flow Act Sum	f_{74}
75	Flow Act Max	f_{75}
76	Flow Act Min	f_{76}
77	Flow Act Mean	f_{77}
78	Flow Act Std	f_{78}
79	Flow Idle Sum	f_{79}
80	Flow Idle Max	f_{80}
81	Flow Idle Min	f_{81}
82	Flow Idle Mean	f_{82}
83	Flow Idle Std	f_{83}
84	Fwd Avg Pkts/Bulk	f_{84}
85	Fwd Avg Bytes/Bulk	f_{85}
86	Fwd Avg Bulk/s	f_{86}
87	Bwd Avg Pkts/Bulk	f_{87}
88	Bwd Avg Bytes/Bulk	f_{88}
89	Bwd Avg Bulk/s	f_{89}

A list of 89 features was introduced, defining them as a set F.

$$F = \{f_1, f_2, \ldots, f_{89}\} \quad (1)$$

The recorded netflow consisted of 64 packet streams. In fact, the number of recorded packets was more, but when combined into a stream, it amounted to 64. After extracting the data recorded from the netflow into packet features, [37] it is checked based on the production rules created to check the network status. The results can be displayed visually and graphically to make the analysis process understandable.

C. Analysis of Network Events

By recording the netflow, packet features were extracted. From these characteristics, the most common and common situations that occur in the network were identified. Fig. 2 shows the situations that occur in the network, divided into seven groups:

- Classic threats;
- Application/Protocol threats;
- Transport - Level threats;
- Hidden/Anomalous traffic;
- Network Anomalous;
- Normal and beneficial flows;
- Flow type.

Table III lists the 110 most common network conditions based on seven groups. This means that the 110 most common network conditions are identified by the characteristics of the extracted packets. The first factor in forming the list of selected threats was the most common, although there may be more such conditions [32]. The second factor in forming the list in Table III was those that can be identified by the characteristics of the packets. Early detection of conditions in the network is always important, which prevents network performance from being disrupted. The characteristics of the extracted packets are used to identify the selected conditions.

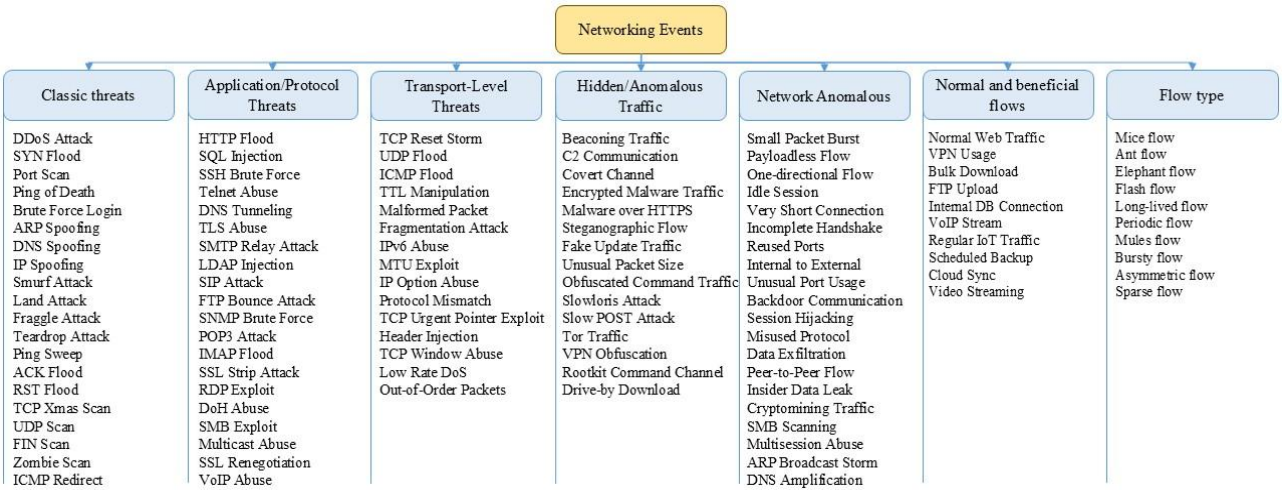


Fig. 2. Grouping of events in the network.

There are various techniques for detecting threats based on package properties. These include mathematical analysis, statistical analysis, visualization, production logic, fuzzy logic, etc [38]. The research used the production logic technique. Production rules serve as a knowledge base for expert systems [39].

TABLE III. THE MOST COMMON NETWORK EVENTS

№	Status name	Marking
1	DDoS Attack	a_1
2	SYN Flood	a_2
3	Port Scan	a_3
4	Ping of Death	a_4
5	Brute Force Login	a_5
6	ARP Spoofing	a_6
7	DNS Spoofing	a_7
8	IP Spoofing	a_8
9	Smurf Attack	a_9
10	Land Attack	a_{10}
11	Fraggle Attack	a_{11}
12	Teardrop Attack	a_{12}
13	Ping Sweep	a_{13}
14	ACK Flood	a_{14}
15	RST Flood	a_{15}
16	TCP Xmas Scan	a_{16}
17	UDP Scan	a_{17}
18	FIN Scan	a_{18}
19	Zombie Scan	a_{19}
20	ICMP Redirect	a_{20}
21	HTTP Flood	a_{21}
22	SQL Injection	a_{22}
23	SSH Brute Force	a_{23}
24	Telnet Abuse	a_{24}
25	DNS Tunneling	a_{25}
26	TLS Abuse	a_{26}
27	SMTP Relay Attack	a_{27}
28	LDAP Injection	a_{28}
29	SIP Attack	a_{29}
30	FTP Bounce Attack	a_{30}
31	SNMP Brute Force	a_{31}
32	POP3 Attack	a_{32}
33	IMAP Flood	a_{33}
34	SSL Strip Attack	a_{34}
35	RDP Exploit	a_{35}
36	DoH Abuse	a_{36}
37	SMB Exploit	a_{37}
38	Multicast Abuse	a_{38}
39	SSL Renegotiation	a_{39}
40	VoIP Abuse	a_{40}
41	TCP Reset Storm	a_{41}
42	UDP Flood	a_{42}
43	ICMP Flood	a_{43}
44	TTL Manipulation	a_{44}
45	Malformed Packet	a_{45}
46	Fragmentation Attack	a_{46}
47	IPv6 Abuse	a_{47}
48	MTU Exploit	a_{48}
49	IP Option Abuse	a_{49}
50	Protocol Mismatch	a_{50}
51	TCP Urgent Pointer Exploit	a_{51}
52	Header Injection	a_{52}
53	TCP Window Abuse	a_{53}
54	Low Rate DoS	a_{54}
55	Out-of-Order Packets	a_{55}
56	Beaconing Traffic	a_{56}
57	C2 Communication	a_{57}
58	Covert Channel	a_{58}
59	Encrypted Malware Traffic	a_{59}
60	Malware over HTTPS	a_{60}
61	Steganographic Flow	a_{61}
62	Fake Update Traffic	a_{62}
63	Unusual Packet Size	a_{63}
64	Obfuscated Command Traffic	a_{64}
65	Slowloris Attack	a_{65}
66	Slow POST Attack	a_{66}
67	Tor Traffic	a_{67}
68	VPN Obfuscation	a_{68}
69	Rootkit Command Channel	a_{69}
70	Drive-by Download	a_{70}
71	Small Packet Burst	a_{71}
72	Payloadless Flow	a_{72}
73	One-directional Flow	a_{73}
74	Idle Session	a_{74}
75	Very Short Connection	a_{75}
76	Incomplete Handshake	a_{76}
77	Reused Ports	a_{77}
78	Internal to External	a_{78}
79	Unusual Port Usage	a_{79}
80	Backdoor Communication	a_{80}
81	Session Hijacking	a_{81}
82	Misused Protocol	a_{82}
83	Data Exfiltration	a_{83}
84	Peer-to-Peer Flow	a_{84}
85	Insider Data Leak	a_{85}
86	Cryptomining Traffic	a_{86}
87	SMB Scanning	a_{87}
88	Multisession Abuse	a_{88}
89	ARP Broadcast Storm	a_{89}
90	DNS Amplification	a_{90}
91	Normal Web Traffic	a_{91}
92	VPN Usage	a_{92}

93	Bulk Download	a_{93}
94	FTP Upload	a_{94}
95	Internal DB Connection	a_{95}
96	VoIP Stream	a_{96}
97	Regular IoT Traffic	a_{97}
98	Scheduled Backup	a_{98}
99	Cloud Sync	a_{99}
100	Video Streaming	a_{100}
101	Mice flow	a_{101}
102	Ant flow	a_{102}
103	Elephant flow	a_{103}
104	Flash flow	a_{104}
105	Long-lived flow	a_{105}
106	Periodic flow	a_{106}
107	Mules flow	a_{107}
108	Bursty flow	a_{108}
109	Asymmetric flow	a_{109}
110	Sparse flow	a_{110}

The network events listed in Table III have been designated as set A.

$$A = \{a_1, a_2, \dots, a_{110}\} \quad (2)$$

The situations recorded in the network status collection are the most frequently detected by network specialists. Network professionals will certainly try to avoid [40] these situations when monitoring network activity and use various software tools for monitoring.

IV. Experimental Results and Discussions

To achieve the goal, it is necessary to automate decision-making using an expert system to analyze netflow traffic and identify faults in it. Based on the netflow traffic, production rules are created for expert systems using the recorded packet features and network events.

A. Formalizing the Mathematical Representation of Production Rules

Production rules are considered as a knowledge base for expert systems. The packet features given in set F are used to identify the 110 network events given in set A. As a result, 110 production rules were created.

Formula 3 is used to create production rules.

$$IF \dots, THEN \dots \quad (3)$$

To put it into mathematical representation, the production rules were written in the form of Boolean rules.

$$p_1: f_{34} \wedge f_{35} \wedge f_{45} \wedge f_{55} \wedge f_{58} \wedge f_{68} \rightarrow a_1 \mid f_{34} > 5000, f_{35} > 10, f_{45} < 1, f_{55} > 100, f_{58} < 10, f_{68} > 50$$

$$p_2: f_{36} \wedge f_{45} \wedge f_{55} \wedge f_{58} \wedge f_{68} \rightarrow a_2 \mid f_{36} > 1000, f_{45} < 1, f_{55} > 100, f_{58} < 5, f_{68} > 30$$

$$p_3: f_7 \wedge f_{33} \wedge f_{55} \rightarrow a_3 \mid f_7 < 5, f_{33} < 200, f_{55} > 20$$

$$p_4: f_6 \wedge f_8 \wedge f_{27} \wedge f_{45} \rightarrow a_4 \mid f_6 = ICMP, f_8 > 10000, f_{27} > 1500, f_{45} > 2$$

$$p_5: f_{36} \wedge f_{55} \wedge f_{58} \rightarrow a_5 \mid f_{36} > 500, f_{55} > 50, f_{85} > 50$$

$$p_6: f_6 \wedge f_7 \wedge f_8 \wedge f_{42} \wedge f_{45} \rightarrow a_6 \mid f_6 = ARP, f_7 = 1, f_8 < 100, f_{42} > 5, f_{45} < 1$$

$$p_7: f_6 \wedge f_7 \wedge f_8 \wedge f_{27} \wedge f_{42} \wedge f_{74} \rightarrow a_7 \mid f_6 = UDP, f_7 < 3, f_8 < 300, f_{27} > 100, f_{42} > 3, f_{74} = 0$$

$$p_8: f_7 \wedge f_8 \wedge f_{42} \wedge f_{55} \wedge f_{58} \wedge f_{74} \rightarrow a_8 \mid f_7 > 50, f_8 > 5000, f_{42} > 20, f_{55} > 50, f_{58} = 0, f_{74} = 0$$

$$p_9: f_6 \wedge f_7 \wedge f_8 \wedge f_{42} \wedge f_{45} \rightarrow a_9 \mid f_6 = ICMP, f_7 > 100, f_8 > 10000, f_{42} > 10, f_{45} < 1$$

$$p_{10}: f_2 \wedge f_3 \wedge f_6 \wedge f_{55} \rightarrow a_{10} \mid f_2 = f_4, f_3 = f_5, f_6 = TCP, f_{55} > 0$$

$$p_{11}: f_6 \wedge f_7 \wedge f_8 \wedge f_{42} \rightarrow a_{11} \mid f_6 = UDP, f_7 > 100, f_8 > 10000, f_{42} > 10$$

$$p_{12}: f_6 \wedge f_8 \wedge f_{12} \wedge f_{27} \rightarrow a_{12} \mid f_6 = TCP, f_8 < 400, f_{12} > 50, f_{27} < 20$$

$$p_{13}: f_6 \wedge f_7 \wedge f_{33} \rightarrow a_{13} \mid f_6 = ICMP, f_7 < 10, f_{33} < 500$$

$$p_{14}: f_{36} \wedge f_{55} \wedge f_{58} \rightarrow a_{14} \mid f_{36} > 500, f_{55} < 10, f_{58} > 50$$

$$p_{15}: f_{36} \wedge f_{56} \wedge f_{58} \rightarrow a_{15} \mid f_{36} > 200, f_{56} > 100, f_{58} < 5$$

$$p_{16}: f_6 \wedge f_{55} \wedge f_{57} \wedge f_{58} \wedge f_{59} \rightarrow a_{16} \mid f_6 = TCP, f_{55} > 0, f_{57} > 0, f_{58} = 0, f_{59} > 0$$

$$p_{17}: f_6 \wedge f_7 \wedge f_8 \wedge f_{36} \rightarrow a_{17} \mid f_6 = UDP, f_7 > 50, f_8 > 5000, f_{36} > 200$$

$$p_{18}: f_6 \wedge f_{36} \wedge f_{55} \wedge f_{56} \wedge f_{58} \wedge f_{59} \rightarrow a_{18} \mid f_6 = TCP, f_{36} > 100, f_{55} = 0, f_{56} = 0, f_{58} = 0, f_{59} = 0$$

$$p_{19}: f_6 \wedge f_7 \wedge f_{33} \wedge f_{58} \rightarrow a_{19} \mid f_6 = TCP, f_7 = 1, f_{33} < 100, f_{58} = 0$$

$$p_{20}: f_6 \wedge f_8 \wedge f_{33} \wedge f_{42} \rightarrow a_{20} \mid f_6 = ICMP, f_8 < 200, f_{33} < 300, f_{42} > 5$$

$$p_{21}: f_5 \wedge f_6 \wedge f_{35} \wedge f_{36} \wedge f_{45} \rightarrow a_{21} \mid f_5 = 80, f_6 = TCP, f_{35} > 10, f_{36} > 1000, f_{45} < 1$$

$$p_{22}: f_5 \wedge f_8 \wedge f_{45} \wedge f_{58} \wedge f_{68} \rightarrow a_{22} \mid f_5 = 80, f_8 > 5000, f_{45} < 1, f_{58} > 50, f_{68} > 50$$

$$p_{23}: f_5 \wedge f_{36} \wedge f_{55} \wedge f_{58} \rightarrow a_{23} \mid f_5 = 22, f_{36} > 500, f_{55} > 50, f_{58} < 5$$

$$p_{24}: f_5 \wedge f_{36} \wedge f_{55} \wedge f_{58} \rightarrow a_{24} \mid f_5 = 23, f_{36} > 200, f_{55} > 30, f_{58} < 10$$

$$p_{25}: f_5 \wedge f_6 \wedge f_8 \wedge f_{27} \rightarrow a_{25} \mid f_5 = 53, f_6 = UDP, f_8 > 2000, f_{27} < 50$$

$$p_{26}: f_5 \wedge f_6 \wedge f_{36} \wedge f_{75} \rightarrow a_{26} \mid f_5 = 443, f_6 = TCP, f_{36} > 300, f_{75} > 1$$

$$p_{27}: f_5 \wedge f_{36} \wedge f_{45} \wedge f_{68} \rightarrow a_{27} \mid f_5 = 25, f_{36} > 100, f_{45} < 2, f_{68} > 20$$

$p_{28}: f_5 \wedge f_8 \wedge f_{36} \wedge f_{58} \rightarrow a_{28} \mid f_5 = 389, f_8 > 3000, f_{36} > 200, f_{58} < 5$

$p_{29}: f_5 \wedge f_7 \wedge f_{36} \rightarrow a_{29} \mid f_5 = 5060, f_7 > 50, f_{36} > 300$

$p_{30}: f_5 \wedge f_7 \wedge f_8 \wedge f_{36} \rightarrow a_{30} \mid f_5 = 21, f_7 > 30, f_8 > 10000, f_{36} > 200$

$p_{31}: f_5 \wedge f_8 \wedge f_{36} \wedge f_{55} \rightarrow a_{31} \mid f_5 = 161, f_8 < 500, f_{36} > 500, f_{55} > 55$

$p_{32}: f_5 \wedge f_{35} \wedge f_{45} \wedge f_{68} \rightarrow a_{32} \mid f_5 = 110, f_{35} > 5, f_{45} < 1, f_{68} > 30$

$p_{33}: f_5 \wedge f_8 \wedge f_{36} \wedge f_{58} \rightarrow a_{33} \mid f_5 = 143, f_8 > 8000, f_{36} > 300, f_{58} < 5$

$p_{34}: f_5 \wedge f_6 \wedge f_{58} \wedge f_{66} \rightarrow a_{34} \mid f_5 = 443, f_6 = TCP, f_{58} > 30, f_{66} < 100$

$p_{35}: f_5 \wedge f_6 \wedge f_{58} \wedge f_{66} \rightarrow a_{35} \mid f_5 = 443, f_6 = TCP, f_{58} > 30, f_{66} < 100$

$p_{36}: f_5 \wedge f_8 \wedge f_{36} \wedge f_{58} \rightarrow a_{36} \mid f_5 = 3389, f_8 > 10000, f_{36} > 500, f_{58} < 5$

$p_{37}: f_5 \wedge f_8 \wedge f_{36} \wedge f_{75} \rightarrow a_{37} \mid f_5 = 443, f_8 > 5000, f_{36} > 200, f_{75} > 1$

$p_{38}: f_4 \wedge f_6 \wedge f_7 \wedge f_{35} \rightarrow a_{38} \mid f_4 = Multicast, f_6 = UDP, f_7 > 100, f_{35} > 5$

$p_{39}: f_5 \wedge f_6 \wedge f_8 \wedge f_{75} \rightarrow a_{39} \mid f_5 = 443, f_6 = TCP, f_8 > 5000, f_{75} > 2$

$p_{40}: f_5 \wedge f_7 \wedge f_{36} \wedge f_{68} \rightarrow a_{40} \mid f_5 = 5060, f_7 > 100, f_{36} > 300, f_{68} > 80$

$p_{41}: f_{36} \wedge f_{56} \wedge f_{58} \wedge f_{74} \rightarrow a_{41} \mid f_{36} > 500, f_{56} > 50, f_{58} < 5, f_{74} = 0$

$p_{42}: f_6 \wedge f_7 \wedge f_8 \wedge f_{42} \wedge f_{45} \rightarrow a_{42} \mid f_6 = UDP, f_7 > 100, f_8 > 8000, f_{42} < 5, f_{45} < 2$

$p_{43}: f_6 \wedge f_7 \wedge f_8 \wedge f_{45} \rightarrow a_{43} \mid f_6 = ICMP, f_7 > 100, f_8 > 10000, f_{45} < 2$

$p_{44}: f_5 \wedge f_6 \wedge f_{27} \wedge f_{42} \rightarrow a_{44} \mid f_5 \neq 0, f_6 = IP, f_{27} > 100, f_{42} < 1$

$p_{45}: f_9 \wedge f_{10} \wedge f_{12} \wedge f_{24} \rightarrow a_{45} \mid f_9 < 20, f_{10} = 0, f_{12} > 50, f_{24} > 30$

$p_{46}: f_5 \wedge f_{12} \wedge f_{18} \wedge f_{33} \rightarrow a_{46} \mid f_5 \neq 0, f_{12} > 70, f_{18} > 30, f_{33} < 200$

$p_{47}: f_6 \wedge f_8 \wedge f_{45} \wedge f_{58} \rightarrow a_{47} \mid f_6 = IPv6, f_8 > 4000, f_{45} > 2, f_{58} < 5$

$p_{48}: f_8 \wedge f_9 \wedge f_{10} \wedge f_{33} \rightarrow a_{48} \mid f_8 > 1400, f_9 > 1300, f_{10} > 10, f_{33} < 300$

$p_{49}: f_6 \wedge f_{25} \wedge f_{27} \wedge f_{74} \rightarrow a_{49} \mid f_6 = IP, f_{25} > 100, f_{27} < 20, f_{74} = 0$

$p_{50}: f_5 \wedge f_6 \wedge f_{42} \wedge f_{58} \rightarrow a_{50} \mid f_5 \neq 25\ f_6 = UDP, f_{42} > 3, f_{58} < 5$

$p_{51}: f_{33} \wedge f_{59} \wedge f_{64} \wedge f_{66} \rightarrow a_{51} \mid f_{33} < 200, f_{59} > 0, f_{64} > 0, f_{66} < 100$

$p_{52}: f_6 \wedge f_{25} \wedge f_{33} \wedge f_{58} \rightarrow a_{52} \mid f_6 = TCP, f_{25} > 100, f_{33} < 300, f_{58} < 10$

$p_{53}: f_{27} \wedge f_{33} \wedge f_{66} \rightarrow a_{53} \mid f_{27} > 1000, f_{33} < 300, f_{66} < 100$

$p_{54}: f_7 \wedge f_{34} \wedge f_{74} \rightarrow a_{54} \mid f_7 > 50, f_{34} < 2, f_{74} > 1000$

$p_{55}: f_{42} \wedge f_{49} \rightarrow a_{55} \mid f_{42} < f_{43}, f_{49} > 50$

$p_{56}: f_{33} \wedge f_{44} \wedge f_{77} \rightarrow a_{56} \mid f_{33} > 60000, f_{44} < 5, f_{77} < 2$

$p_{57}: f_5 \wedge f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{57} \mid f_5 \neq 80, f_7 < 10, f_8 < 1000, f_{33} > 30000$

$p_{58}: f_5 \wedge f_8 \wedge f_{48} \wedge f_{74} \rightarrow a_{58} \mid f_5 \neq 80, f_8 < 500, f_{48} < 3, f_{74} > 1000$

$p_{59}: f_5 \wedge f_6 \wedge f_8 \wedge f_{68} \rightarrow a_{59} \mid f_5 = 443, f_6 = TCP, f_8 > 5000, f_{68} > 20$

$p_{60}: f_5 \wedge f_{68} \wedge f_{74} \rightarrow a_{60} \mid f_5 = 443, f_{68} > 30, f_{74} > 5000$

$p_{61}: f_8 \wedge f_9 \wedge f_{45} \wedge f_{74} \rightarrow a_{61} \mid f_8 < 800, f_9 < 100, f_{45} > 20 f_{74} > 1000$

$p_{62}: f_5 \wedge f_8 \wedge f_{74} \wedge f_{77} \rightarrow a_{62} \mid f_5 = 80, f_8 < 500, f_{74} < 200, f_{77} < 2$

$p_{63}: f_9 \wedge f_{10} \wedge f_{12} \rightarrow a_{63} \mid f_9 > 1400, f_{10} < 50, f_{12} > 200$

$p_{64}: f_6 \wedge f_8 \wedge f_{58} \wedge f_{74} \rightarrow a_{64} \mid f_6 = TCP, f_8 < 400, f_{58} < 3, f_{74} > 2000$

$p_{65}: f_{33} \wedge f_{58} \wedge f_{68} \wedge f_{69} \rightarrow a_{65} \mid f_{33} > 10000, f_{58} < 3, f_{68} = 0, f_{69} = 0$

$p_{66}: f_7 \wedge f_{58} \wedge f_{68} \rightarrow a_{66} \mid f_7 > 50, f_{58} < 3, f_{68} = 0$

$p_{67}: f_5 \wedge f_8 \wedge f_{44} \rightarrow a_{67} \mid f_5 \in \{9001; 443\}, f_8 > 2000, f_{44} > 10$

$p_{68}: f_5 \wedge f_6 \wedge f_8 \wedge f_{34} \rightarrow a_{68} \mid f_5 \in \{1194; 443\}, f_6 = TCP, f_8 > 4000, f_{34} < 10$

$p_{69}: f_6 \wedge f_8 \wedge f_{58} \wedge f_{74} \rightarrow a_{69} \mid f_6 = TCP, f_8 < 800, f_{58} < 2, f_{74} > 3000$

$p_{70}: f_5 \wedge f_8 \wedge f_{74} \wedge f_{77} \rightarrow a_{70} \mid f_5 = 80, f_8 < 2000, f_{74} < 1000, f_{77} < 1$

$p_{71}: f_7 \wedge f_9 \wedge f_{33} \rightarrow a_{71} \mid f_7 > 30, f_9 < 100, f_{33} < 1000$

$p_{72}: f_8 \wedge f_{68} \wedge f_{69} \rightarrow a_{72} \mid f_8 = 0, f_{68} = 0, f_{69} = 0$

$p_{73}: f_{13} \vee f_{19} \rightarrow a_{73} \mid f_{13} = 0, f_{19} = 0$

$p_{74}: f_8 \wedge f_{33} \wedge f_{74} \rightarrow a_{74} \mid f_8 < 500, f_{33} > 300000, f_{74} < 50$

$p_{75}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{75} \mid f_7 < 5, f_8 < 200, f_{33} < 200$

$p_{76}: f_{54} \wedge f_{55} \wedge f_{58} \rightarrow a_{76} \mid f_{54} = 0, f_{55} = 1, f_{58} = 0$

$p_{77}: f_5 \wedge f_8 \wedge f_{66} \rightarrow a_{77} \mid f_5 = f_3, f_8 > 0, f_{66} < 500$

$p_{78}: f_2 \wedge f_4 \rightarrow a_{78} \mid f_2 = 19.168.\%, f_4 \neq 192.168.\%$

$p_{79}: f_5 \wedge f_8 \rightarrow a_{79} \mid f_5 \notin \{80,443,53,21,22,25\}, f_8 > 1000$

$p_{80}: f_5 \wedge f_8 \wedge f_{33} \wedge f_{36} \wedge f_{66} \rightarrow a_{80} \mid f_5 \in \{4444; 1337\}, f_8 > 0, f_{33} > 10000, f_{36} < 1, f_{66} < 1024$

$p_{81}: f_8 \wedge f_{33} \wedge f_{55} \wedge f_{56} \wedge f_{58} \rightarrow a_{81} \mid f_8 > 500, f_{33} < 500, f_{55} = 1, f_{56} = 1, f_{58} = 1$

$p_{82}: f_6 \wedge f_8 \rightarrow a_{82} \mid f_6 \notin \{1; 6; 17\}, f_8 > 1000$

$p_{83}: f_2 \wedge f_4 \wedge f_8 \wedge f_{36} \rightarrow a_{83} \mid f_2 = 192.168.\%, f_4 \neq 192.168.\%, f_8 > 5000, f_{36} > 5$

$p_{84}: f_5 \wedge f_8 \wedge f_{36} \rightarrow a_{84} \mid f_5 \in \{6881; 51413; 135\}, f_8 > 2000, f_{36} > 10$

$p_{85}: f_2 \wedge f_8 \wedge f_{33} \rightarrow a_{85} \mid f_2 = f_4, f_8 > 2000, f_{33} < 1000$

$p_{86}: f_5 \wedge f_6 \wedge f_8 \wedge f_{34} \wedge f_{66} \rightarrow a_{86} \mid f_5 = 3333, f_6 = 6, f_8 > 5000, f_{34} > 10, f_{66} > 2000$

$p_{87}: f_5 \wedge f_{36} \wedge f_{68} \rightarrow a_{87} \mid f_5 = 445, f_{36} > 5, f_{68} > 10$

$p_{88}: f_{33} \wedge f_{70} \wedge f_{72} \rightarrow a_{88} \mid f_{33} < 300, f_{70} > 10, f_{72} > 10$

$p_{89}: f_6 \wedge f_{33} \wedge f_{34} \rightarrow a_{89} \mid f_6 = 1, f_{33} < 100, f_{34} > 50$

$p_{90}: f_5 \wedge f_8 \wedge f_{20} \wedge f_{38} \rightarrow a_{90} \mid f_5 = 53, f_8 > 512, f_{20} > 2000, f_{38} > 3$

$p_{91}: f_5 \wedge f_8 \wedge f_{36} \rightarrow a_{91} \mid f_5 \in \{80; 443\}, f_8 > 1000, f_{36} > 1$

$p_{92}: f_5 \wedge f_{36} \wedge f_{66} \rightarrow a_{92} \mid f_5 \in \{1194; 443\}, f_{36} < 1, f_{66} > 10000$

$p_{93}: f_5 \wedge f_8 \wedge f_{36} \rightarrow a_{93} \mid f_5 = 80, f_8 > 100000, f_{36} > 10$

$p_{94}: f_5 \wedge f_8 \wedge f_{13} \rightarrow a_{94} \mid f_5 = 21, f_8 > 10000, f_{13} > 10$

$p_{95}: f_5 \wedge f_{13} \wedge f_{14} \rightarrow a_{95} \mid f_5 = 3306, f_{13} > 5, f_{14} > 1000$

$p_{96}: f_5 \wedge f_{14} \wedge f_{36} \rightarrow a_{96} \mid f_5 = 5060, f_{14} < 200, f_{36} > 10$

$p_{97}: f_5 \wedge f_8 \wedge f_{36} \rightarrow a_{97} \mid f_5 \in \{1883; 8883\}, f_8 < 1000, f_{36} < 2$

$p_{98}: f_5 \wedge f_8 \wedge f_{33} \rightarrow a_{98} \mid f_5 \in \{445; 139\}, f_8 > 50000, f_{33} > 100000$

$p_{99}: f_5 \wedge f_8 \wedge f_{33} \wedge f_{36} \rightarrow a_{99} \mid f_5 \in \{80; 443\}, f_8 > 20000, f_{33} > 60000, f_{36} > 2$

$p_{100}: f_5 \wedge f_8 \wedge f_{33} \wedge f_{36} \rightarrow a_{100} \mid f_5 = 443, f_8 > 500000, f_{33} > 100000, f_{36} > 5$

$p_{101}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{101} \mid f_7 < 5, f_8 < 100, f_{33} < 500$

$p_{102}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{102} \mid f_7 < 3, f_8 < 60, f_{33} < 100$

$p_{103}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{103} \mid f_7 > 1000, f_8 > 5000000, f_{33} > 100000$

$p_{104}: f_8 \wedge f_{33} \wedge f_{36} \rightarrow a_{104} \mid f_8 < 500, f_{33} < 100, f_{36} > 20$

$p_{105}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{105} \mid f_7 > 100, f_8 > 100000, f_{33} > 300000$

$p_{106}: f_7 \wedge f_{82} \wedge f_{83} \rightarrow a_{106} \mid f_7 > 10, f_{82} < 20000, f_{83} < 2000$

$p_{107}: f_7 \wedge f_8 \wedge f_{33} \rightarrow a_{107} \mid f_7 > 20, 10000 < f_8 < 50000, 2000 < f_{36} < 20000$

$p_{108}: f_8 \wedge f_{33} \wedge f_{78} \rightarrow a_{108} \mid f_8 > 500, f_{33} < 2000, f_{78} > 1000$

$p_{109}: f_8 \wedge f_{19} \wedge f_{40} \rightarrow a_{109} \mid f_8 > 1000, f_{19} = 0, f_{40} > 20$

$p_{110}: f_8 \wedge f_{33} \wedge f_{36} \rightarrow a_{110} \mid f_8 < 500, f_{33} > 100000, f_{36} < 1$

The created rules serve as a knowledge base for expert systems. The conditions in the rules were determined taking into account the characteristics and features of the network event being analyzed. The definition of the generated production rules as a set P has been introduced.

$$P = \{p_1, p_2, \dots, p_{110}\} \quad (4)$$

So $P: F \rightarrow A$ is expressed as a set.

B. Define Conditions for Production Rules

Not all of the extracted packet features were used in the mathematical representation of production rules. Because only the most optimal ones were selected to determine the network events. However, it is possible to express production rules using all features. However, only the optimal features were selected to eliminate redundant computations and improve the algorithm.

When determining the conditions of the production rule, the conditions were determined based on the characteristics of the network event. The 110 productions developed were derived from rule-defining network events in specifying rule conditions. That is, each rule was formed based on the characteristics of the network condition. In defining these conditions, the most desirable packet features were selected.

C. The Most Important Production Rules

From the perspective of network flow analysis, 25 production rules were selected as the most important, frequently occurring, and important to identify.

$$P = \{p_1, p_2, p_3, p_5, p_6, p_{19}, p_{22}, p_{25}, p_{26}, p_{42}, p_{43}, p_{58}, p_{60}, p_{64}, p_{65}, p_{67}, p_{69}, p_{72}, p_{80}, p_{81}, p_{84}, p_{85}, p_{86}, p_{90}, p_{103}\}$$

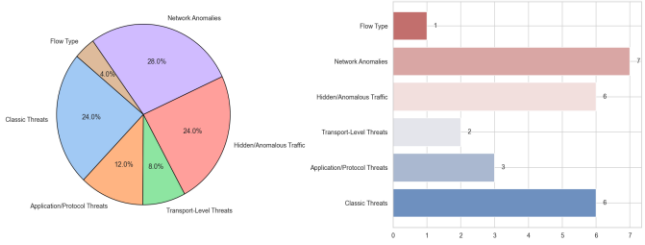


Fig. 3. The most important production rule group.

Among the selected production rules, there is no network event that falls exclusively into the group of normal and useful flows. The reason is that, firstly, the selected productions are aimed at the efficient operation of the system and the non-destructive operation of the system. Secondly, the events that fall into this group are often considered normal flows and are allowed in the network. However, they are used for analysis and filtering. Therefore, the 25 most priority production rules were

formed due to their priority and frequency of occurrence in the selected production rules. Fig. 3 shows the selected production rules divided into groups.

A netflow dataset recorded in a laboratory environment was tested against 25 selected production rules.

Algorithm 4 Testing production rules

```
1: Load packet data from "packet_features.csv"
2: Initialize prediction column to empty string
3: Define apply_rules(row):
4:     if f_34 > 5000 & f_35 > 10 & f_45 < 1 & f_55 >100
       & f_58 < 10 & f_68 > 50:
5:         return "a1"
6:     else if f_36 > 1000 & f_45 < 1 & f_55 > 100 &
       f_58 < 5 & f_68 > 30:
7:         return "a2"
8:     else if f_7 < 5 & f_33 < 200 & f_55 > 20:
9:         return "a3"
10:    ...
11:    [continue similarly with all rule conditions up
       to a103]
12:    ...
13:    else:
14:        return " "
15: for each row in dataset:
16:    apply apply_rules to assign value to Prediction
       column
17: save updated dataset to "prediction_output.csv"
```

The results obtained from the dataset, which was checked based on the priority production rules, were stored in the "prediction_output" file (see Algorithm 4).

Algorithm 5 Production rules analysis

```
1: Load result data from "prediction_output.csv"
2: extract values from 'Prediction' column that are not null
   or "Normal"
3: remove duplicates and convert to list → worked_rules
4: define priority_rules as the list of 25 important rule
   identifiers:
    ['a1','a2','a3','a5','a6','a19','a22','a25','a26','a42','a43',
     'a58','a60','a64','a65','a67','a69','a72','a80','a81',
     'a84','a85','a86','a90','a103']
5: compute not_worked_rules = priority_rules-
   worked_rules
6: print number and list of worked_rules (sorted)
7: print number and list of not_worked_rules (sorted)
```

To analyze the tested dataset, we analyzed the production rules that worked and the production rules that did not work (see Algorithm 5).

D. Results Obtained Based on Production Rules

The result of the check can be seen in Fig. 4. When checking the dataset, not all of the selected top priority rules worked. Because all the given conditions must be met for all the rules to work. So, $a_1, a_2, a_{22}, a_{43}, a_{60}, a_{90}, a_{103}$ network events were detected, and the production rules $p_1, p_2, p_{22}, p_{43}, p_{60}, p_{90}, p_{103}$ worked, respectively. Out of 64 netflows, 50 normal flows and 14 corresponding to the network event were detected.

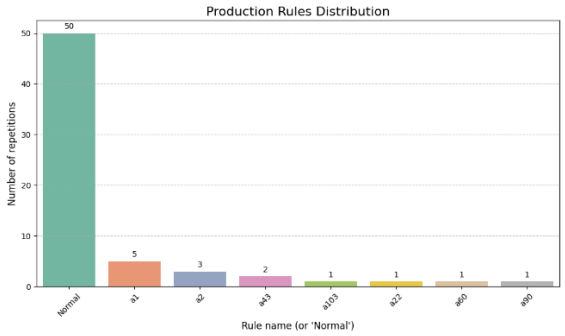


Fig. 4. Production rules analysis result.

The result obtained is presented in Fig. 5 in the form of a distribution of normal and working production rules.

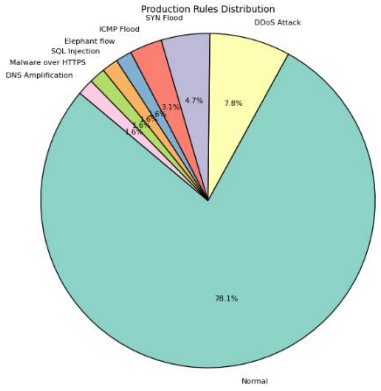


Fig. 5. Netflow general analysis.

The results of the network flow analysis recorded based on the developed production rules are presented in Fig. 6. Using the confusion matrix, the following results were obtained:

$$True\ Positive = 43, \quad False\ Positive = 2,$$

$$False\ Negative = 7, \quad True\ Negative = 12.$$

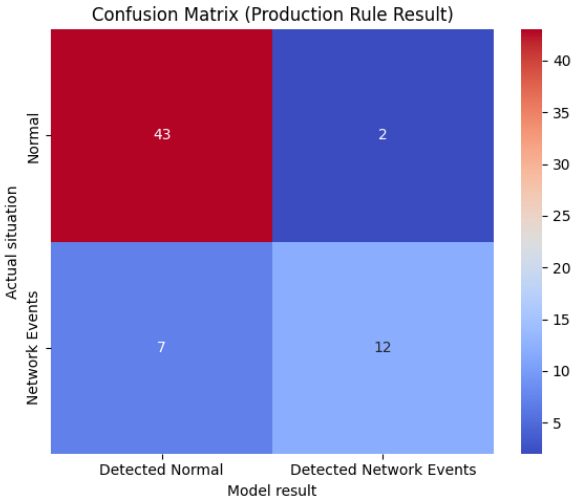


Fig. 6. Confusion matrix of production rules.

The network status detection methodology developed based on production rules achieved accuracy 85.94%, precision 95.56%, recall 86.00% and F1-score 90.53%. The results are presented in Table IV and Fig. 7.

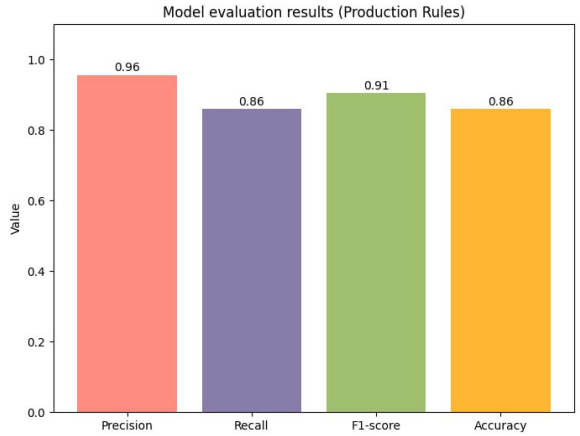


Fig. 7. Graph of model evaluation results.

TABLE IV. EVALUATION OF THE RESULT OF PRODUCTION RULES

№	Evaluation	Result
1	Accurancy	85,94%
2	Precision	95.56%
3	Recall	86.00%
4	F1-score	90.53%

The result obtained based on the ROC curve in Fig. 9 was 0.86, indicating that the developed production rules could effectively distinguish the network events that could occur. According to the Precision-Recall Curve results in Fig. 8, the average accuracy of the system was 0.92.

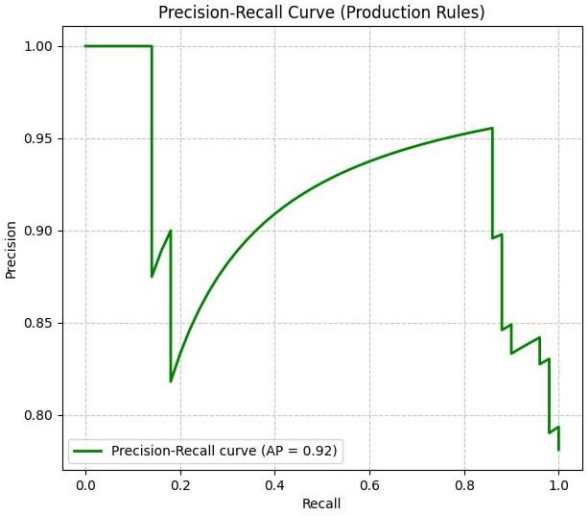


Fig. 8. Graph of precision-recall curve evaluation result.

This further confirms the reliability of the model in identifying attack flows. Thus, the developed rule-based model can make correct decisions in most of the netflows, but it is desirable to extend and refine the rules to increase sensitivity.

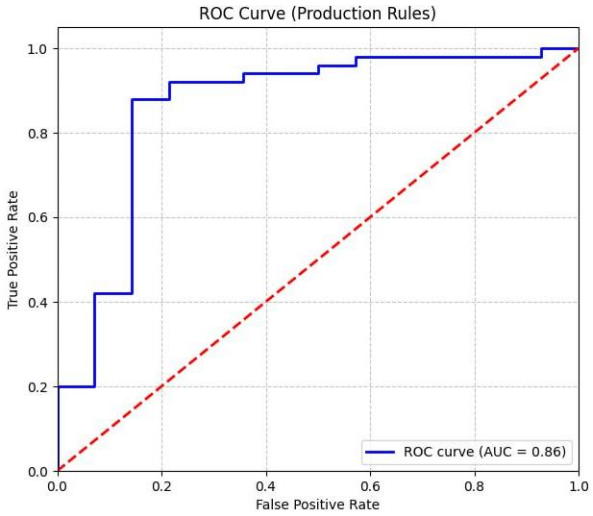


Fig. 9. Graph of ROC curve evaluation result.

V. CONCLUSION AND FUTURE WORKS

In automated and digital form, expert research results were presented in NetFlow monitoring analysis, mathematical representation in production logic decision-making. Real-time processing and collection of NetFlow traffic events in cyberspace were programmed. NetFlow data was converted to CSV and collected in PCAP format by performing a three-step algorithmic sequence. In a classical theoretical-fundamental way, cyber threats, cyber threats at application stages, anomalies and simple digital data flows were presented. These features identified 110 frequent network events and synthesized them into seven sets. Each event was formalized by the implementation of Boolean rules. These production rules were organized into a knowledge base that supports optimal identification of cyber threats. The protection of the infrastructure and the effective organization of the network events identification in the digital data flow were provided by architecture and analysis. A proposed decision tree based on Production logic was supported by fusion analysis.

At the same time, we plan to expand the work studied above in our future research. To do this, first of all, we want to analyze new unexplored areas of the industry in the implementation of a new scientific innovation project funded by the Ministry of Higher Education, Science and Innovation of the Republic of Uzbekistan.

REFERENCES

[1] Yarashov, I., Rakhimberdiev, K., Jumabayeva, A., Subhonov, M., & Oydinoy, K. (2024, December). Modelling the Process of Ensuring Information Security in Inclusive Education Platforms Based on Functional Tables. In Proceedings of the 8th International Conference on Future Networks & Distributed Systems (pp. 881-889).

[2] Yarashov, I. (2022, September). Development of a reliable method for grouping users in user access control based on a Functioning table. In 2022 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-5). IEEE.

[3] Toshmatov, S., Yarashov, I., Otakhonov, A., & Ismatillayev, A. (2022, September). Designing an algorithmic formalization of threat actions based on a Functioning table. In 2022 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-5). IEEE.

[4] A. Kabulov, I. Saymanov, I. Yarashov and A. Karimov, "Using Algorithmic Modeling to Control User Access Based on Functioning Table," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795850.

[5] Kabulov, A.; Baizhumanov, A.; Saymanov, I. Synthesis of Optimal Correction Functions in the Class of Disjunctive Normal Forms. Mathematics 2024, 12, 2120. https://doi.org/10.3390/math12132120

[6] Kabulov, A., Baizhumanov, A., Berdimurodov, M. (2024). On the minimization of k-valued logic functions in the class of disjunctive normal forms. Journal of Mathematics, Mechanics and Computer Science, 121(1), 37–45. https://doi.org/10.26577/JMMCS202412114

[7] Sikos L. F. Knowledge representation to support partially automated honeypot analysis based on Wireshark packet capture files //Intelligent Decision Technologies 2019: Proceedings of the 11th KES International Conference on Intelligent Decision Technologies (KES-IDT 2019), Volume 1. – Singapore : Springer Singapore, 2019. – С. 345-351.

[8] Kabulov, A., & Yarashov, I. (2021, November). Mathematical model of information processing in the ecological monitoring information system. In 2021 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-4). IEEE.

[9] Kabulov, A.; Saymanov, I.; Babadjanov, A.; Babadzhanov, A. Algebraic Recognition Approach in IoT Ecosystem. Mathematics 2024, 12, 1086. https://doi.org/10.3390/math12071086

[10] Saymanov, I. (2024). Logical automatic implementation of steganographic coding algorithms. Journal of Mathematics, Mechanics and Computer Science, 121(1), 122–131. https://doi.org/10.26577/JMMCS2024121112

[11] Kabulov, A.; Normatov, I.; Saymanov, I.; Baizhumanov, A. On the Completeness of Classes of Correcting Functions of Heuristic Algorithms, Azerbaijan Journal of Mathematics, 2025, vol 15, no. 2. https://doi.org/10.59849/2218-6816.2025.2.51

[12] A. Kabulov, I. Normatov, A. Seytov and A. Kudaybergenov, "Optimal Management of Water Resources in Large Main Canals with Cascade Pumping Stations," 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2020, pp. 1-4, doi: 10.1109/IEMTRONICS51293.2020.9216402.

[13] Boateng A., Rahim M. M. Statistical analysis of network data flows and predictions using statistical and machine learning regression models : дис. – Brac University, 2024.

[14] Kabulov, A. V., Normatov, I. H., Ashurov A.O. Computational methods of minimization of multiple functions. Journal of Physics: Conference Series, 1260(10), 10200, 2019. doi:10.1088/1742-6596/1260/10/102007.

[15] Islambek Saymanov, et al. Numerical Methods of Synthesis of a Correct Algorithm for Solving Recognition Problems. Advances in Artificial Intelligence and Machine Learning. 2025;5(1):202. https://dx.doi.org/10.54364/AAIML.2025.51202

[16] Normatov, I., Yarashov, I., Otakhonov, A., & Ergashev, B. (2022, September). Construction of reliable well distribution functions based on the principle of invariance for convenient user access control. In 2022 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-5). IEEE.

[17] Maruf Juraev, Inomjon Yarashov, Adilbay Kudaybergenov, Alimdzhan Babadzhanov and Zilolaxon Mamatova, “Research on Network Flow Based on Statistical Analysis Methods” International Journal of Advanced Computer Science and Applications(ijacsa), 16(6), 2025. http://dx.doi.org/10.14569/IJACSA.2025.0160618

[18] A. Kabulov, I. Yarashov and A. Otakhonov, "Algorithmic Analysis of the System Based on the Functioning Table and Information Security," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795746.

[19] E. Navruzov and A. Kabulov, "Detection and analysis types of DDoS attack," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2022, pp. 1-7, doi: 10.1109/IEMTRONICS55184.2022.9795729.

[20] Batista, E., Alencar, B., Silva, E., Canário, J., Rios, R. A., Dustdar, S., . . . Prazeres, C. (2024). A new intelligent scheduler to improve reactive OpenFlow communication in SDN-based IoT data streams. Discover Internet of Things, 4(1), 15. doi:https://doi.org/10.1007/s43926-024-00068-3

[21] Yarashov, I., Juraev, M., Ismatillayev, A., & Rakhimberdiev, K. (2024, December). Behavior Modeling in Computerized Production Systems. In Proceedings of the 8th International Conference on Future Networks & Distributed Systems (pp. 873-880).

[22] Sikos, Leslie F. "Packet analysis for network forensics: A comprehensive survey." Forensic Science International: Digital Investigation 32 (2020): 200892.

[23] Laptiev, O., Musienko, A., Nakonechnyi, V., Sobchuk, A., Gakhov, S., & Kopytko, S. (2023, June). Algorithm for recognition of network traffic anomalies based on artificial intelligence. In 2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-5). IEEE.

[24] Carlet, Claude. "Boolean functions." Encyclopedia of Cryptography, Security and Privacy. Cham: Springer Nature Switzerland, 2025. 292-296.

[25] Yao, Yuxiang, Zi-Gang Huang, and Duanqing Pei. "Diversified dynamic effects and their order origins in Boolean functions." Chaos, Solitons & Fractals 191 (2025): 115830.

[26] Carlet, C., Propagation Characteristics of Boolean Functions. In Encyclopedia of Cryptography, Security and Privacy (pp. 1984-1985). (2025).

[27] Kellerer, N., Sánchez, G., Alberto, H., Hagenmeyer, V., & Elbez, G., Attacks on the Siemens S7 Protocol Using an Industrial Control System Testbed. In Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems (pp. 770-779). (2025, June).

[28] Ţălu, M., DDoS Mitigation in Kubernetes: A Review of ExtendedBerkeley Packet Filtering and eXpress Data Path Technologies. JUTI: Jurnal Ilmiah Teknologi Informasi, 60-73. (2025).

[29] Bittencourt, G., & Marengoni, M., A customizable tool for the generation of production-based systems. WIT Transactions on Information and Communication Technologies, 2. (2025).

[30] Rahmatullah, R., Gao, H., Utama, R. P., Adi, P. D. P., Mubashir, J., Muwardi, R., ... & Dwiyanti, H., Multinode LoRa-MQTT of Design Architecture and Analyze Performance for Dual Protocol Network IoT. International Journal of Advanced Computer Science & Applications, 16(1). (2025).

[31] Li, X., & Chen, X. (2025). Intelligent Identification of Pile Defects Based on Improved LSTM Model and Wavelet Packet Local Peaking Method. International Journal of Advanced Computer Science & Applications, 16(5).

[32] Jing, G. U. O., Dejun, Z. H. U., & Qing, X. U. (2025). Metaheuristic-Driven Feature Selection for IoT Intrusion Detection: A Hierarchical Arithmetic Optimization Approach. International Journal of Advanced Computer Science & Applications, 16(6).

[33] Tahboush, M., Hamdan, A., Klaib, M., Adawy, M., & Alzobi, F. (2025). Detection Optimization of Brute-Force Cyberattack Using Modified Caesar Cipher Algorithm Based on Binary Codes (MCBC). International Journal of Advanced Computer Science & Applications, 16(3).

[34] Guo, L., Yan, F., Li, T., Yang, T., & Lu, Y. (2022). An automatic method for constructing machining process knowledge base from knowledge graph. Robotics and Computer-Integrated Manufacturing, 73, 102222.

[35] Yarashov, I. (2021, November). Algorithmic Formalization Of User Access To The Ecological Monitoring Information System. In 2021 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-3). IEEE.

[36] Yarashov, I., Kuvonchbek, R., Juraev, M., & Rahmonov, F. (2024, December). Processing and Analysis of Network Flow Data Using Intelligent Technologies. In Conference on Internet of Things and Smart Spaces (pp. 40-51). Cham: Springer Nature Switzerland.

[37] Normatov, I., Yarashov, I., & Toshmatov, S. (2024). Research and modeling of authentication process using functioning table. Journal of Mathematics, Mechanics and Computer Science, 124(4), 71-85.

[38] Kabulov, A. V., Normatov, I. H. About problems of decoding and searching for the maximum upper zero of discrete monotone functions. Journal of Physics: Conference Series, 1260(10), 102006, 2019. doi:10.1088/1742-6596/1260/10/102006.

[39] A. Kabulov, I. Normatov, E. Urunbaev and F. Muhammadiev, "Invariant Continuation of Discrete Multi-Valued Functions and Their Implementation," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1-6, doi: 10.1109/IEMTRONICS52119.2021.9422486.

[40] A. Kabulov, I. Saymanov, I. Yarashov and F. Muxammadiev, "Algorithmic method of security of the Internet of Things based on steganographic coding," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2021, pp. 1-5, doi: 10.1109/IEMTRONICS52119.2021.9422588.

AI-Enabled Demand Forecasting, Technological Capability, and Supply Chain Performance: Empirical Evidence from the Global Logistics Sector

Mohamed Amine Frikha[1], Mariem Mrad[2]
King Faisal University, Applied College, Saudi Arabia[1]
Faculty of Economics and Management of Sfax, Tunisia[2]

***Abstract*—This study advances understanding of artificial intelligence (AI) integration within supply chain management, with a particular emphasis on AI-enabled demand forecasting. The research examines 1) the extent of adoption of AI-driven forecasting practices, 2) the role of technological and organizational readiness, captured through data infrastructure, workforce skills, and management support, as antecedents, and 3) the mediating effect of AI adoption on the relationship between readiness and supply chain performance. Grounded in the resource-based view and technology adoption theory, a conceptual model was developed and empirically validated using data from global logistics firms, with structural equation modeling applied as the primary analytical technique. The findings confirm that readiness factors significantly foster AI adoption, which in turn exerts both a direct effect on supply chain performance and a mediating effect linking readiness to performance. By focusing on the global logistics sector and empirically validating this mediating mechanism, the study provides novel insights into how firms can translate technological readiness into superior operational outcomes, offering theoretical contributions to AI assimilation literature and practical guidance for managers.**

Keywords*—AI-enabled demand forecasting; technological capability; data infrastructure; workforce skills; management support; supply chain performance; artificial intelligence

I. Introduction

Over the past decade, the configuration of global supply chains has been subject to profound and unprecedented changes, reshaping their operational dynamics and strategic orientation. Driven by fluctuating market demands, geopolitical uncertainties, the rapid growth of e-commerce, and the increasing need for advanced data-driven insights, the economic environment remains in constant flux, prompting organizations to adapt swiftly and effectively to these evolving dynamics. In this context, forecasting based on AI-enabled intelligence has emerged as a critical determinant of success and a strategic lever for sustaining competitive advantage [1]. The central challenge is not to justify the benefits of AI in forecasting or to assess its limitations, but to define the prerequisites for its effective deployment and the crucial mechanisms that facilitate its adoption.

Scholars and industry practitioners have increasingly aligned around a growing momentum toward AI-based strategies, particularly with the advancement of machine learning and predictive analytics. As a robust application of contemporary technologies, AI-powered demand forecasting is gaining recognition as a leading method for managing supply chains with elevated efficiency and performance [2]. AI has thus evolved into a vital tool in the domain of supply chain management [3].

The interest in exploring this specific research area underscores the criticality of a core business function. Indeed, several academics [4] argue that despite AI's increasing relevance, the literature still lacks a comprehensive understanding of how companies can effectively shift from conventional techniques to AI-based forecasting models. While recent contributions have illuminated the drivers and consequences of AI integration on business outcomes [5], there remains an evident gap regarding how to successfully innovate under the paradigm of AI-driven forecasting. In addressing this challenge, the present study posits that understanding the value of AI in supply chain contexts requires not only an analysis of its precursors and its direct effects on firm performance but also an exploration of its mediating function in linking organizational preparedness to performance outcomes.

As part of corporate integration within an evolving knowledge-driven ecosystem, a firm's technological and organizational preparedness has frequently been acknowledged in scholarly works as a fundamental condition for successful digital transformation. Aligned with this view, the technology-organization-environment framework posits that in order to fully capitalize on emerging technologies, firms must cultivate internal competencies to enable the effective assimilation of such innovations [6]. Accordingly, examining supply chain effectiveness in the context of AI implementation through the lens of organizational readiness has become both a managerial imperative and a vibrant field of inquiry.

This article pursues three main objectives. First, it seeks to assess the extent of AI-enabled forecasting adoption based on technological and organizational preparedness. Second, it evaluates the direct influence of AI utilization on supply chain outcomes. Third, it investigates more intricate relationships by analyzing the mediating role of AI integration between organizational readiness and performance, as advocated in prior research [7]. Specifically, this study aims to capture the indirect effects of internal readiness on organizational outcomes via the intermediary of AI adoption. This emerging research stream, which integrates the resource-based view with

AI assimilation frameworks, is regarded as particularly relevant for conceptualizing and empirically verifying novel relationships that capture the multifaceted dimensions of the phenomenon. Accordingly, the contributions of this study are threefold: 1) it examines the global logistics sector, an important but under-investigated context for AI adoption; 2) it consolidates multiple readiness factors into a single integrative model; and 3) it empirically demonstrates the mediating role of AI adoption in translating readiness into superior supply chain performance.

To address these objectives, a conceptual model was developed and empirically assessed through the testing of multiple hypotheses. The remainder of the paper is structured as follows. Section II reviews the literature and formulates the research hypotheses. Section III describes the methodology, including data collection, measurement, and analytical strategy. Section IV presents the results, while Section V discusses theoretical and managerial implications. Section VI concludes with limitations and avenues for future research.

II. Literature Review and Research Hypotheses

During the last ten years, AI-enabled forecasting has become a focal point in supply chain management discourse. The model is founded on the assumption that firms can, and are encouraged to, employ sophisticated analytical techniques across both internal and external data sources to enhance the accuracy of their forecasts. From a data-centric organizational standpoint, the AI model encourages the integration of advanced technologies, including machine learning, predictive analytics, and generative adversarial networks (GANs) [8].

By AI-enabled demand forecasting, we aim the activities of leveraging advanced algorithms on large datasets, grouped according to three major technologies identified through an extensive literature review: predictive analytics using machine learning, deep learning models for time-series analysis, and advanced simulation tools such as generative AI. Consistent with prevailing research in the field, this study highlights these technologies by investigating a new research direction that has the potential to advance current knowledge [9]. Specifically, we examine the level of AI adoption by assessing its determinants, technological and organizational readiness, and their impact on firm performance. In doing so, two approaches have been advocated to better understand success in the modern supply chain: the technology adoption approach and the resource-based view. We specify three more specific objectives on which we rely to structure our discourse. First, we show the impact of readiness on the adoption of AI [10]. Then, we assess the effect of AI adoption on supply chain performance. Finally, we propose the articulation between readiness and performance through the mediation of AI practices.

A. The Impact of Technological and Organizational Capabilities on AI Adoption

According to existing research, there is a broad consensus that forecasting from an AI perspective would require the firm to possess certain resources and a state of readiness to ensure the better adoption of these complex practices and to take full advantage of them [11]. In resource-based theory, the primary object of research consists of building and regenerating competitive advantages. In this respect, a firm's internal resources and state of readiness dictate its ability to adapt to an ever-changing environment. For this research, we identify three readiness factors that are extensively supported in the literature as key drivers facilitating the adoption of new and complex technologies: quality, workforce skills and AI literacy, management support, strategic alignment and data infrastructure.

1) The impact of data infrastructure and quality on the degree of AI adoption: Academic discussions on AI implementation emphasize the necessity of a robust data infrastructure for organizations looking to leverage advanced analytics [12]. Data quality, accessibility, and integration constitute critical sources of competitive advantage and are essential for the effective application of AI principles. AI algorithms, particularly machine learning, are fundamentally dependent on large volumes of high-quality, structured data. Without a solid data foundation, the outputs of these models are unreliable, hindering adoption. Challenges such as data silos and poor data quality are consistently cited as major barriers to AI implementation [13]. Conversely, firms that have invested in data warehouses, data lakes, and robust ETL (Extract, Transform, Load) processes are better positioned to experiment with and deploy AI models successfully. Therefore, a mature data infrastructure would favor the strategic decision-making process and lead to the successful implementation of AI forecasting practices.

Accordingly, this study seeks to test the relationship between organizational data infrastructure and the extent of AI adoption, as stated in the following research hypothesis:

- H1a - A robust data infrastructure and high data quality foster the degree of adoption of AI-enabled demand forecasting practices.

2) The impact of workforce skills and AI literacy on the degree of AI adoption: A second key determinant of organizational readiness, consistently supported in the literature, is the firm's human capital, particularly the AI literacy and competencies of its workforce. This concept is well established and closely relates to absorptive capacity [14], defined as the ability to identify, assimilate, and apply new knowledge. In AI research, considerable emphasis has been placed on the critical role of a skilled workforce for successful implementation [15]. Organizations aiming to adopt AI must not only acquire the technology but also have internal expertise to manage, interpret, and leverage its outputs. To date, most studies have identified the "skills gap" as a major barrier [16]. We therefore posit that a skilled workforce, comprising data scientists, analysts, and AI-savvy managers, empowers firms to effectively recognize and harness the value of AI applications. Furthermore, the ability to integrate the outputs of AI models into existing business processes is essential to take full advantage of their virtues. To

empirically examine the linkage between this capability and AI adoption, we propose the second sub-hypothesis.

- H1b - Workforce skills and AI literacy foster the degree of adoption of AI-enabled demand forecasting practices.

3) The impact of management support and strategic alignment on the degree of AI adoption: Management support and strategic alignment refer to the firm's ability to provide resources for AI initiatives. In a context of technological change, this support is a central element for firms seeking to innovate. To better succeed in an AI adoption approach, companies must show commitment from leadership. The literature highlights that a key challenge for managers during AI implementation is securing sufficient budget and aligning the initiative with broader business objectives [15]. Strong executive sponsorship can significantly enhance the extent of AI adoption and ensure sustained investment by mitigating concerns over resource reallocation. Moreover, the presence of a clear strategic vision for AI can be marked by its integration into the company's long-term plan. This strategic alignment ensures that AI projects are not isolated experiments but are tied to core business objectives, such as cost reduction or market share growth. This vision seems to be an important factor in the performance of an AI strategy by enabling the company to properly value its investments and show stakeholders the importance of these new capabilities. Despite growing interest, empirical research on the direct relationship between management support and the extent of AI adoption remains limited. To address this gap, we propose to empirically examine this relationship through the formulation of the third sub-hypothesis:

- H1c - Management support and strategic alignment foster the degree of adoption of AI-enabled demand forecasting practices.

B. AI Adoption and Supply Chain Performance

In supply chain management, numerous studies highlight the critical role of AI, arguing that different AI techniques can markedly boost organizational performance. This relationship is supported by three main factors. The rationale underpinning this affirmative correlation is predicated upon three principal factors. Primarily, enterprises integrating AI in their forecasting processes benefit from a significant increase in accuracy due to the ability of algorithms to analyze vast and complex datasets [17]. This improved accuracy directly reduces inventory holding costs and the risk of stockouts. Secondly, the literature on AI in supply chains emphasizes automation and efficiency. Engaging AI for forecasting automates a complex and time-consuming task, freeing up human capital for more strategic activities [18]. Consequently, this leads to a reduction in costs associated with manual forecasting. As Sajja et al. (2025) argue, in today's highly competitive and rapidly evolving environment, companies are increasingly adopting AI-driven approaches to mitigate cost and risk challenges. Ultimately, by leveraging AI-enabled forecasting, firms can improve their service levels and overall operational performance. These practices enable the supply chain to be more responsive and consequently improve customer satisfaction with better product availability [19].

However, in reviewing the literature, we found that large-scale empirical studies remained limited until recently. Since the early 2020s, studies have begun to robustly examine the effect of AI on the performance of supply chains. Hasan et al. (2024) demonstrated that AI applications are positively associated with improvements in operational efficiency and innovation performance. Parida et al. (2012), in a parallel context, examined the impact of inbound openness on performance, reporting positive effects. More recently, studies by Singh (2025) and Boualam et al. (2025) concluded that AI adoption promotes internal efficiency and consequently gives the company a competitive advantage and superior performance. The results of their empirical examinations indicated that AI-driven intelligence has a strong effect on collaboration and firm performance [20-21].

Based on these theoretical and empirical findings, we aim to examine the positive relationship between the extent of AI practice adoption and supply chain performance. Consequently, we propose the following research hypothesis:

- H2: The degree of adoption of AI-enabled demand forecasting practices improves supply chain performance.

C. The Role of AI Adoption as a Mediator Between Readiness and Performance

Existing studies consistently emphasize the critical role of a company's preparedness in establishing a competitive edge. This preparedness is regarded as a pathway for businesses to develop and enhance new abilities, enabling them to make effective use of their resources. Numerous researchers have leveraged this idea to illustrate its influence on the pursuit of innovative initiatives and, ultimately, on overall business success.

Consequently, in environments marked by scarce resources and fluctuating markets, organizations depend on their technical and operational preparedness to prioritize the discovery and application of cutting-edge, AI-supported strategic technologies. Utilizing their data frameworks [6], staff expertise [11], and executive endorsement, leaders must promote the smooth integration of these advancements into internal planning systems through AI-driven demand prediction techniques, which are broadly acknowledged as crucial drivers of performance.

Building on the analysis presented, this study perceives supply chain effectiveness as the product of efforts to unify resources and expertise. As a result, companies may improve their outcomes more efficiently by capitalizing on their preparedness, which empowers them to adopt emerging technologies, recognize market possibilities, and harness new insights. Nevertheless, there remains a scarcity of hands-on research exploring the link between preparedness and performance within an AI framework. This gap makes it compelling to explore how AI adoption serves as a bridge between these preparedness components and organizational

success. Therefore, the following hypotheses regarding mediating relationships are proposed:

- H3a: The degree of AI adoption mediates the relationship between data infrastructure and firm performance.
- H3b: The degree of AI adoption mediates the relationship between workforce skills and firm performance.
- H3c: The degree of AI adoption mediates the relationship between management support and firm performance.

III. Methodology

In light of the preceding developments, this research seeks to assess the level of adoption of AI-enabled demand forecasting practices as shaped by technological and organizational readiness factors. Additionally, it evaluates the subsequent impact of this adoption on supply chain performance. The conceptual model depicted in Fig. 1 articulates the underlying research problem and provides the basis for empirical validation.

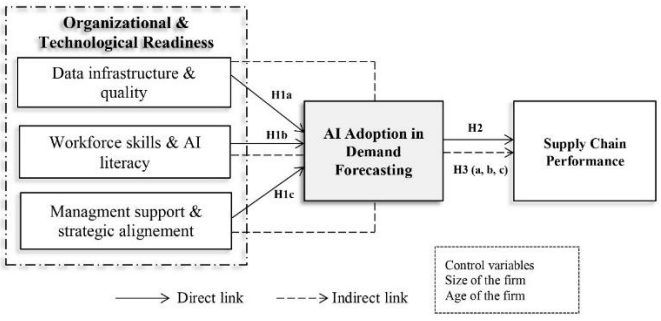


Fig. 1. Conceptual framework illustrating AI adoption's mediating role between readiness and supply chain performance.

A. Model Specification

To fulfill the research objectives, this study adopts a quantitative approach to empirically evaluate the proposed hypotheses. Data were gathered through a field survey administered to firms within the global logistics industry. The following sections elaborate on the sampling strategy, data collection process, and the measurement of key variables.

The following regression equations describe the direct effects model used to test hypotheses H1a to H2:

$$AI_ADOPT = \beta_0 + \beta_1(DATA_INFRA) + \beta_2(SKILLS) + \beta_3(SUPPORT) + \beta_4(SIZE) + \beta_5(AGE) + \varepsilon_1 \quad (1)$$

$$PERF = \gamma_0 + \gamma_1(AI_ADOPT) + \gamma_2(SIZE) + \gamma_3(AGE) + \varepsilon_2 \quad (2)$$

Where:

- AI_ADOPT is the degree of AI Adoption.
- PERF is the Supply Chain Performance.
- DATA_INFRA, SKILLS, and SUPPORT are the organizational readiness factors.
- SIZE and AGE are control variables.
- β_0 and γ_0 are the intercepts.
- $\beta_1 ... \beta_5$ and $\gamma_1 ... \gamma_3$ are the regression coefficients to be calculated.
- ε_1 and ε_2 are the error terms.

B. Data Collection Procedure and Sample Design

In this empirical investigation, data were collected using a structured questionnaire, selected for its effectiveness in capturing standardized responses across a large sample. Particular attention was given to the clarity and precision of the questionnaire items to ensure unambiguous interpretation by respondents.

The sampling process was primarily based on a global directory of logistics and supply chain management firms, supplemented by purposive outreach to industry professionals through platforms such as LinkedIn. Following a non-probabilistic convenience sampling approach, and in alignment with practices commonly adopted in logistics research, a total of 235 valid responses were obtained. Data were collected through multiple channels, electronic distribution, postal mailings, telephone interviews, and in-person administration, to enhance participation. This multi-modal strategy resulted in a response rate of approximately 41%, which is consistent with empirical studies in the field.

C. Measures of the Variables

Ensuring construct validity and reliability, the variables were defined through a comprehensive literature review and measured using multi-item instruments on a 5-point Likert scale.

1) Technological and organizational readiness: This was operationalized through three distinct constructs:

a) Data Infrastructure and Quality (DATA_INFRA): We selected six items based on the literature concerning data readiness for AI. These items assessed: the accessibility of data across the organization, the quality and accuracy of historical data, the integration of data from various sources (sales, marketing, external factors), data security protocols, the scalability of data storage, and the availability of real-time data streams.

b) Workforce Skills and AI Literacy (SKILLS): We refer to the operationalization of absorptive capacity [19] adapted to an AI context. Four dimensions were chosen: the capacity to attract and retain talent with data science skills (acquisition); the ability of staff to understand and interpret AI model outputs (assimilation); the capacity to integrate AI insights into daily operational workflows (transformation); and the ability to leverage AI for commercial and strategic purposes (exploitation).

c) Management Support and Strategic Alignment (SUPPORT): We selected three items reflecting leadership commitment: the capacity of management to champion and fund AI projects; the existence of a clear, strategic roadmap for AI integration; and the alignment of AI initiatives with core business objectives (cost reduction, improving customer

satisfaction). In the questionnaire, firms were asked to specify the degree of their capacity for each item using a Likert scale from 1 "very limited capacity" to 5 "very extensive capacity".

- Degree of AI Adoption (AI_ADOPT): Our model presents this as a mediating variable. For operationalization, we selected ten items based on common AI applications in forecasting, drawing from recent work [16]: Three items associated with the *use of external data sources* for forecasting (market trends, economic indicators, weather data); four items associated with *specific AI techniques* (use of machine learning models, use of deep learning for time-series, use of predictive analytics for what-if scenarios, use of AI for real-time demand sensing); and three items associated with the *integration of AI outputs* (use of AI for automated inventory replenishment, use of AI to inform logistics planning, and use of AI to generate reports for management. In the questionnaire, companies were asked to indicate both their level of use and the perceived importance of these various applications. Each item was measured on a Likert scale ranging from 1 ("not at all important") to 5 ("extremely important").
- Supply Chain Performance (PERF): This dependent variable was measured using an indicator reflecting the degree of satisfaction with nine elements measuring overall supply chain performance. These elements were chosen based on pioneering work [4] and refer respectively to: forecast accuracy improvement; inventory turnover rates; reduction in stockout incidents; On-Time In-Full delivery rates; reduction in logistics and holding costs; lead time reduction; overall operational efficiency; customer satisfaction levels; and overall profitability of the supply chain. In the questionnaire, the degree of satisfaction with each item is specified according to the perception of company managers using a scale from 1 "very unsatisfactory" to 5 "very satisfactory".
- Control Variables: We also introduce in our model two control variables: the size (number of employees) and age of the firm.

D. Data Analysis Strategy

To analyze the collected data and validate the proposed research model, a three-stage statistical procedure was implemented. First, the unidimensionality of the measurement scales was assessed using Principal Component Analysis (PCA) with Varimax rotation, to confirm that each construct was adequately captured by a coherent set of indicators. This step was essential for ensuring the structural clarity of the latent variables. Second, reliability analyses were performed through the computation of Cronbach's alpha coefficients, verifying the internal consistency of each multi-item scale. These preliminary analyses were conducted using SPSS software. In the third phase, the hypotheses concerning the structural relationships among the variables were tested through Structural Equation Modeling (SEM), using the AMOS software package and the maximum likelihood estimation (MLE) method. To assess the overall model fit, we adopted the classification of fit indices proposed by Roussel et al. (2002), which includes absolute fit indices (χ^2/df, RMSEA), incremental fit indices (CFI, TLI), and parsimony fit indices (PNFI). Once model adequacy was established, the structural paths were assessed based on three main indicators: 1) the standardized regression weights, which indicate the relative strength of the relationships; 2) the critical ratio (CR) values, which test the statistical significance of each path; and 3) the sign of the coefficients, to confirm the expected directionality of the effects. Finally, to evaluate the mediating role of AI adoption in the relationship between organizational readiness and supply chain performance, we employed the procedure outlined by Baron and Kenny (1986), which enabled us to determine whether the mediation effect was statistically significant.

IV. Results

A. Reliability and Validity Analysis

To evaluate the unidimensionality and reliability of the measurement constructs, Table I presents the results of the PCA and internal consistency analysis. Each construct yielded a single dominant factor with a high proportion of total variance explained, ranging from 80.004% (AI Adoption) to 91.580% (Workforce Skills), confirming the unidimensionality of the measurement scales. The KMO values for all constructs exceed 0.80, with highly significant Bartlett's tests (p=0.000), indicating the suitability of the data for factor analysis. No items were removed during the analysis. Additionally, the reliability of the scales is supported by strong Cronbach's alpha coefficients, ranging from 0.879 (AI Adoption) to 0.937 (Data Infrastructure), demonstrating excellent internal consistency across all constructs.

TABLE I. Principal Component Analysis and Reliability

Variables Analysed	Number of items	Extracted Factors	KMO Indice	Total variance explained	Cronbach's Alpha
Data Infrastructure	6	DATA_INFRA	0.923 (0.000)*	84.110	0.937
Workforce Skills	4	SKILLS	0.875 (0.000) *	91.580	0.899
Management Support	3	SUPPORT	0.811 (0.000) *	82.002	0.905
AI Adoption	10	AI_ADOPT	0.828 (0.000) *	80.004	0.879
Supply Chain Performance	9	PERF	0.838 (0.000) *	81.002	0.892

* Bartlett Sphericity Test (Sig)

The confirmation of unidimensionality and the validation of the measurement scales provided a solid foundation for testing the proposed conceptual model. The outcomes of the hypothesis tests are detailed in the following section.

B. Verification and Validation of the Hypotheses

The results of testing our conceptual model represented in Fig. 2 and Tables II and III.

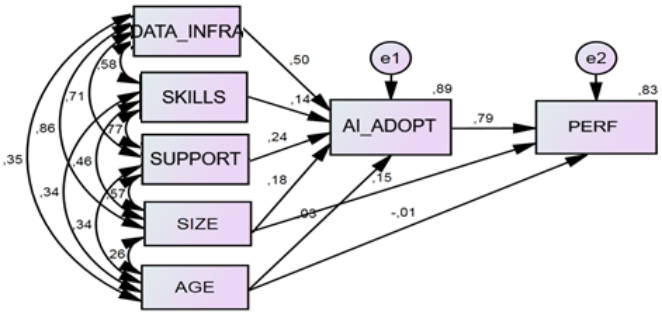


Fig. 2. Structural model – direct effects analysis.

To evaluate the direct relationships among variables in the research model (Fig. 2), the results summarized in Table II demonstrate a satisfactory model fit, providing robust empirical support for the hypothesized structural relationships.

TABLE II. FIT INDICES FOR THE STRUCTURAL MODEL ASSESSING DIRECT EFFECTS

Absolute indices			Incremental indices			Parsimony indices		
GFI	*AGFI*	*RMSEA*	*NFI*	*CFI*	*TLI*	*DMIN/DF*	*AIC*	*CAIC*
0.995	0.961	0.017	0.998	1.000	0.999	3.216/3 = 1.072 *P*=0.351	55.021	165.732

TABLE III. RESULTS OF DIRECT EFFECTS ANALYSIS USING STRUCTURAL EQUATION MODELING

Structural relations	Structural parameters		
	Direct effects	*CR*	*Sig. P*
DATA_INFRA → AI_ADOPT	0.496	9.704	0.000
SKILLS → AI_ADOPT	0.145	4.201	0.000
SUPPORT → AI_ADOPT	0.236	5.972	0.000
SIZE → AI_ADOPT	0.174	4.065	0.000
AGE → AI_ADOPT	0.030	1.212	0.226
AI_ADOPT → PERF	0.789	16.400	0.000
SIZE → PERF	0.152	3.285	0.001
AGE → PERF	-0.006	-0.207	0.836

As presented in Table III, the SEM results provide empirical evidence on the direct associations between organizational enablers and the adoption of AI within firms. Among the examined factors, data infrastructure emerges as the most influential determinant (β = 0.496; p = 0.000), highlighting the pivotal role of digital capabilities in facilitating AI integration. In parallel, both employee skillsets (β = 0.145; p = 0.000) and executive support (β = 0.236; p = 0.000) exhibit significant and positive effects on AI adoption, though their relative impacts are less substantial compared to that of data infrastructure. These findings provide empirical support for hypotheses H1a, H1b, and H1c, indicating that these internal organizational factors are critical drivers of AI adoption.

Regarding the control variables, the analysis reveals that firm size exerts a statistically significant positive effect on AI adoption (β = 0.174; CR = 4.065; p = 0.000), indicating that larger firms may possess greater resources and capabilities to invest in advanced technologies. In contrast, firm age does not appear to have a significant influence (β = 0.030; CR = 1.212; p = 0.226), suggesting that organizational maturity alone is insufficient to drive AI implementation.

Furthermore, the model validates a strong and significant relationship between AI adoption and firm performance (β = 0.789; CR = 16.400; p = 0.001), thereby supporting hypothesis H2. This result highlights the strategic relevance of AI integration as a performance-enhancing factor in contemporary organizational settings. The positive and significant effect of firm size on performance (β = 0.152; CR = 3.285; p = 0.001) reinforces this trend, whereas firm age remains non-significant (β = –0.006; CR = –0.207; p = 0.836), indicating that innovation outcomes are more likely to be associated with firm scalability rather than longevity.

To examine the presence of indirect effects, a second structural model was estimated following the mediation analysis framework proposed by Baron and Kenny (1986). This model was designed to evaluate both the direct and indirect pathways connecting dynamic organizational capabilities to innovation performance, incorporating the degree of openness to inbound open innovation practices as a mediating variable. The conceptual framework of this mediation model is depicted in Fig. 3.

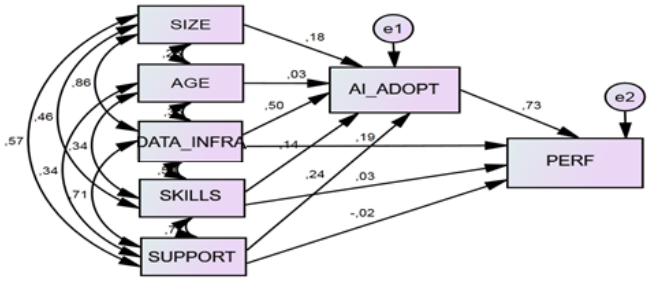


Fig. 3. Structural model assessing the mediating role of AI adoption.

Table IV presents the model fit indices associated with the mediation analysis, which confirm the adequacy of the model and support its suitability for representing the data structure.

TABLE IV. CONCEPTUAL FRAMEWORK FOR THE MEDIATING ROLE OF AI ADOPTION

Absolute indices			incremental indices			Parsimony indices		
GFI	*AGFI*	*RMSEA*	*NFI*	*CFI*	*TLI*	*DMIN/DF*	*AIC*	*CAIC*
0.994	0.910	0.084	0.997	0.998	0.979	5.191/2 = 2.595 P=0.075	57.192	172.355

Table V summarizes the direct and indirect effects derived from the mediation model, which examines how organizational

enablers influence firm performance through the intermediary role of AI adoption. The preceding analyses confirmed that data infrastructure, employee skills, and top management support have significant positive effects on the adoption of AI technologies. In parallel, the findings revealed a robust and statistically significant association between AI adoption and organizational performance, thereby supporting Hypothesis H2.

To explore the mediating dynamics more comprehensively, a second model was estimated using AMOS, allowing for simultaneous evaluation of both direct and indirect paths. In this expanded framework, AI adoption remains a robust predictor of performance (β = 0.726; CR = 9.092; p = 0.000), confirming its central role as a mediating construct in the relationship between internal organizational factors and performance outcomes.

The estimates shown in Table V indicate that data infrastructure continues to exhibit a direct and statistically significant influence on performance (β = 0.195; p = 0.003), in addition to a notable indirect effect (β = 0.360) via AI adoption. This pattern is consistent with partial mediation, wherein the effect of data infrastructure is only partially channeled through the mediator. Conversely, the direct effects of skills (β = 0.027; p = 0.547) and support (β = –0.016; p = 0.757) on performance are no longer statistically significant when accounting for the mediating role of AI adoption. These findings point to full mediation, suggesting that the influence of these two factors on performance operates entirely through the adoption of AI technologies.

TABLE V. DIRECT AND INDIRECT RELATIONSHIPS BY STRUCTURAL EQUATION MODELING

Structural relations	Structural Parameters			
	direct Effects (β)	*CR*	*Sig. P*	*indirect Effects (β)*
DATA_INFRA → PERF	0.195	2.967	0.003	0.360
SKILLS → PERF	0.27	0.602	0.547	0.105
SUPPORT → PERF	-0.016	-0.310	0.757	0.171
AI_ADOPT → PERF	**0.726**	9.092	0.000	-

The results show that in the presence of the mediator (AI_ADOPT), the direct effects of Workforce Skills and Management Support on Supply Chain Performance become insignificant (p=0.547 and p=0.757, respectively). This indicates perfect mediation. However, the direct relationship between Data Infrastructure and Supply Chain Performance remains significant (β=0.195, p=0.003), indicating partial mediation.

We can therefore confirm hypothesis H3a through the presence of partial mediation and hypotheses H3b and H3c through perfect mediation. This confirms that organizational readiness factors, Data Infrastructure, Workforce Skills, and Management Support, have a significant relationship with supply chain performance indirectly through the degree of adoption of AI-enabled demand forecasting.

V. DISCUSSION

This study sought to examine the factors influencing supply chain performance by integrating the Resource-Based View with the Technology Adoption Theory. By empirically examining both direct and mediated relationships, the analysis contributes to a deeper understanding of how organizational readiness factors, Such as data infrastructure, workforce skills, and top management support, influence AI adoption and, in turn, supply chain performance. In addition to validating direct linkages, the results deliver compelling evidence confirming AI adoption as a significant mediator in the proposed relationship framework.

Specifically, the statistical analysis reveals that technological and organizational readiness have a positive impact on the level of AI adoption, which subsequently exerts a strong and significant influence on supply chain performance. These outcomes are consistent with prior theoretical and empirical research [9-11]. For example, global logistics firms equipped with mature data infrastructures tend to adopt AI-enabled forecasting tools more proactively. The observation of partial mediation in the case of data infrastructure suggests a dual mechanism: while a solid technological foundation facilitates AI implementation, it may also directly enhance performance outcomes, potentially through improved reporting systems, real-time analytics, or other non-AI digital capabilities.

The results also indicate that skills and managerial support are fully mediated by AI adoption in their relationship with performance. This finding implies that the presence of skilled personnel or supportive leadership, while necessary, is not sufficient to improve supply chain outcomes in isolation. Their true value is realized when these capabilities are mobilized to support and implement AI technologies effectively. In this sense, skills and support function as enablers of technological transformation, with AI adoption serving as the operative mechanism that translates readiness into performance gains.

By empirically validating the proposed conceptual model, this study advances the existing literature on AI deployment within supply chain management. The analytical approach, focusing on both direct and mediating effects, adds methodological and theoretical value, offering a nuanced view of how internal organizational capabilities interact with emerging digital technologies. This multi-layered perspective on causality is relatively novel in supply chain research, where mediation models remain underexplored.

From a managerial standpoint, the findings underscore the importance of building AI readiness across multiple dimensions. Executives should recognize that the mere acquisition of AI tools does not guarantee performance improvements. Instead, value creation from AI depends on foundational investments in data systems, workforce development, and sustained leadership commitment. These factors collectively determine whether AI adoption efforts yield tangible returns in operational performance.

VI. Conclusion

This study demonstrates that organizational readiness, captured through data infrastructure, workforce skills, and top management support, exerts its influence on supply chain performance primarily through the adoption of AI-enabled demand forecasting. While data infrastructure shows both direct and mediated effects, skills and managerial support improve performance only indirectly via AI adoption.

These findings highlight the pivotal role of AI adoption as the mechanism translating organizational capabilities into operational performance gains. For practitioners, the results stress the need for holistic readiness, encompassing not only technological investments but also human capital and leadership commitment.

Despite its contributions, this study is subject to certain limitations. First, the exclusive focus on demand forecasting as the examined AI application may restrict the breadth and generalizability of the findings across other supply chain functions. To enhance external validity, future research should investigate additional AI-driven applications, such as inventory optimization, predictive maintenance, and warehouse automation, that may reveal complementary or divergent patterns of mediation. Moreover, the current model primarily emphasizes internal readiness factors, overlooking external contingencies. Future studies would benefit from incorporating environmental dimensions, including market competition, industry maturity, and regulatory frameworks, to provide a more holistic understanding of AI adoption and its performance implications.

Acknowledgment

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. 253225].

References

[1] R. M. Baron and D. A. Kenny, "The moderator-mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations," *J. Pers. Soc. Psychol.*, vol. 51, no. 6, p. 1173, 1986.

[2] M. Boualam, I. I. El Farouk, and F. Jawab, "Revolutionizing the Demand-Driven Supply Chain: AI and Machine Learning Applications," in *Supply Chain Transformation Through Generative AI and Machine Learning*. IGI Global, 2025.

[3] M. A. Frikha and Y. Boujelbene, "The importance of the logistics information system 'LIS' for the facilitation of customs control: case of authorized economic operators 'AEO'," in *Proc. 2024 IEEE 15th Int. Colloq. Logistics and Supply Chain Management (LOGISTIQUA)*, Sousse, Tunisia, May 2024.

[4] W. M. Cohen and D. A. Levinthal, "Absorptive capacity: A new perspective on learning and innovation," *Adm. Sci. Q.*, vol. 35, pp. 128–152, 1990.

[5] K. Danach, A. E. Dirani, and H. Rkein, "Revolutionizing Supply Chain Management With AI: A Path to Efficiency and Sustainability," *IEEE Access*, 2024.

[6] S. R. P. Dandamudi, J. Sajja, and A. Khanna, "Advancing cybersecurity and data networking through machine learning-driven prediction models," *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 13, no. 1, pp. 26–33, 2025.

[7] S. S. Goswami, S. Mondal, S. Sarkar, and R. Halder, "Artificial Intelligence-Enabled Supply Chain Management: Unlocking New Opportunities and Challenges," in *Artificial Intelligence and Applications*, 2025.

[8] L. M. Gutta, B. Dhamodharan, P. K. Dutta, and P. Whig, "AI-infused quantum machine learning for enhanced supply chain forecasting," in *Quantum Computing and Supply Chain Management: A New Era of Optimization*. IGI Global, 2024.

[9] K. S. Harish, P. U. Kotehal, M. M. Sandesh, and G. R. Lokesh, "Artificial Intelligence in Supply Chain Management: Trends and Implications," *Nanotechnol. Percept.*, 2024.

[10] M. Z. Hasan, M. Z. Hussain, S. Umair, and U. Waqas, "Understanding Artificial Intelligence in Supply Chain and Innovation Performance," in *Human Perspectives of Industry 4.0 Organizations: Reviewing Sustainable Performance*. CRC Press, 2024.

[11] E. K. R. E. Huizingh, "Open innovation: State of the art and future perspectives," *Technovation*, vol. 31, no. 1, pp. 2–9, 2011.

[12] R. Krishnan, P. Sandhiya, and S. Marwah, "Reskilling Supply Chain Workforce for AI-Driven Service Marketing to Enhance Efficiency and Customer Engagement," in *Strategic Workforce Reskilling in Service Marketing*, 2025.

[13] T. T. H. Nguyen, "Applications of Artificial Intelligence for Demand Forecasting," *Oper. Supply Chain Manag.*, 2023.

[14] V. Parida, M. Westerberg, and J. Frishammar, "Inbound open innovation activities in high-tech SMEs: The impact on innovation performance," *J. Small Bus. Manag.*, vol. 50, no. 2, pp. 283–309, 2012.

[15] S. Popa, P. Soto-Acosta, and I. Martinez-Conesa, "Antecedents, moderators, and outcomes of innovation climate and open innovation: An empirical study in SMEs," *Technol. Forecast. Soc. Change*, vol. 118, pp. 134–142, 2017.

[16] S. Sağ, B. Sezen, and M. Güzel, "Factors that motivate or prevent adoption of open innovation by SMEs in developing countries and policy suggestions," *Procedia - Soc. Behav. Sci.*, vol. 235, no. 2016, pp. 756–763, 2016.

[17] B. Singh, "Revolutionizing Supply Chains for Optimized Demand Planning, Inventory Management, and Logistics: An In-Depth Analysis of AI and ML Solutions in the Modern Era," in *Supply Chain Transformation Through Generative AI and Machine Learning*. IGI Global, 2025.

[18] L. Thirupathi, S. H. Sunkara, I. R. Boodala, and V. S. Vempati, "Personalization and customization strategies with AI in service marketing," in *Integrating AI-Driven Technologies Into Service Marketing*, 2024.

[19] O. Zabraoui, A. Chafi, and S. K. Alami, "Integrating Artificial Intelligence in Supply Chain: Enhancements and Challenges," in *2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology, IRASET 2025*, 2025.

[20] S. Zahra and G. George, "Absorptive capacity: A review, reconceptualization, and extension," *Acad. Manag. Rev.*, vol. 27, no. 2, pp. 185–203, 2002.

[21] M. Mrad, M. A. Frikha, and Y. Boujelbene, "A Comprehensive Survey of Artificial Intelligence and Robotics for Reducing Carbon Emissions in Supply Chain Management," *Logistics*, vol. 9, no. 3, p. 104, Aug. 2025, doi: 10.3390/logistics9030104.

A Starfish Optimization Algorithm-Based Federated Learning Approach for Financial Risk Prediction in Manufacturing Enterprises

Bin Liu[1], Liang Chen[2], Haitong Jiang[3]*, Rui Ma[4]
School of Economics and Management, Aba Teachers College, Aba, China[1]
School of Economics and Management, Sichuan Vocational College of Chemical Technology, Liuzhou, China[2]
School of Culture Management, Sichuan Vocational College of Culture and Communication, Chengdu, China[3]
Health and Rehabilitation College, Chengdu University of Traditional Chinese Medicine, Chengdu, China[4]

***Abstract*—During digital transformation, manufacturing enterprises encounter challenges such as the high cost of smart devices, operational interruptions, and increased technology expenses, raising their financial risks. Addressing the digital transformation challenges confronting manufacturing enterprises necessitates developing an intelligent financial risk prediction system leveraging AI technologies like big data and deep learning, enabling enterprises to mitigate financial exposure. In addition, some data of some manufacturing enterprises cannot be disclosed and shared due to the involvement of trade secrets and shareholder interests. To address these challenges, this study proposes a federated learning (FL)-based framework for predicting financial risk in manufacturing enterprises. Without sharing data, each client (manufacturing enterprise) in the FL framework uses deep learning models to train financial risk prediction models through a central server federation. In this study, the proposed FL framework employs a deep learning model based on a neural Turing machine (NTM) with a long short-term memory (LSTM) controller. In addition, in order to improve the prediction accuracy of the hybrid NTM-FL model, an improved starfish optimization algorithm (ISFOA) was used to optimize the structure of the NTM model. Finally, the experimental results showed that the ISFOA-based NTM-FL (ISFOA-NTM-FL) model improved the prediction accuracy by 26.32% compared to the other three financial risk prediction models.**

***Keywords*—Deep learning; neural Turing machine; prediction; starfish optimization algorithm; federated learning**

I. Introduction

The intelligent financial risk management system is of great significance to manufacturing enterprises by enabling accurate risk prediction and optimized fund allocation. However, their digital transformation introduces critical financial risk challenges. These include high costs of smart devices, investment risks brought by technological iterations, and substantial capital expenditures that can constrain cash flow [1]. In addition, the massive capital expenditures during the digital transformation of manufacturing enterprises may also lead to cash flow constraints. Furthermore, the pace of technological change can quickly render previous investments ineffective, failing to meet profit expectations. Traditional financial risk prediction models struggle to accurately quantify these specific manufacturing risks [2]. To address this challenge, manufacturers are actively pursuing the development of AI-based intelligent financial risk management systems to more effectively manage financial risks throughout their transformation and ongoing operations.

Recurrent neural network (RNN), as a fundamental architecture in the field of deep learning, focuses on feature extraction and complex pattern recognition through multi-layer nonlinear transformations [3]. However, traditional RNN models have significant limitations in handling sequence dependent tasks, with one of the biggest challenges being long-term memory decay. Indicative examples include long short-term memory (LSTM) networks and gate recurrent units (GRU) [4]. The neural Turing machine (NTM), as a typical model under the deep learning architecture, introduces external memory on the basis of the deep learning infrastructure and constructs a hybrid computing system with active memory read and write capabilities. Compared with LSTM and GRU models, the introduction of external memory enables the NTM model to have memory function and achieve information reuse across time steps. Therefore, another typical application of NTM is time series prediction that considers causal relationships. For example, when processing quarterly financial reports of manufacturing enterprises, NTM can autonomously associate the causal relationship between equipment investment and current profitability of manufacturing enterprises, while traditional RNNs can only indirectly capture such relationships through fixed window time series feature engineering. Therefore, in the field of financial risk prediction, NTM has disruptive potential.

In addition, the development of financial risk prediction models for manufacturing enterprises faces severe data privacy challenges. The operational data of manufacturing enterprises involves a large amount of core business data such as inventory costs and supply chain costs, so the financial risk prediction of manufacturing enterprises has the characteristics of strong data sensitivity and large data volume [5]. At the same time, when manufacturing enterprises complete digital transformation, the above data will be scattered in independent systems such as manufacturing management systems and supply chain management systems, making integration difficult. The traditional RNN framework requires manufacturing enterprises to upload a large amount of raw data to third-party platforms

* Corresponding author.

during model training, which poses a risk of privacy data leakage for the enterprise. These risks may lead to market share losses and cost structure leaks for manufacturing companies. In summary, intelligent manufacturing enterprises often hold multiple private data that cannot be shared, which has led to a decrease in the accuracy of financial risk prediction in traditional models.

Federated learning (FL) technology has the potential to protect data privacy in financial risk prediction for manufacturing enterprises through distributed computing frameworks. Specifically, FL adopts a multi-layer federated encryption protocol, which can isolate the financial data of various manufacturing enterprises in a local encryption domain. Sensitive fields such as research investment cost, equipment investment cost, and supply chain cost of manufacturing enterprises are encrypted and trained locally with relevant model parameters, which are then uploaded to the global model of the alliance. Therefore, this study focuses on the financial risk prediction problem of manufacturing enterprises, considering the complexity and privacy constraints of manufacturing enterprise data. Based on the NTM model and FL framework, a hybrid NTM-FL framework is designed, which uses the NTM model with an LSTM controller to predict the financial risk of manufacturing enterprises. At the same time, the improved starfish optimization algorithm (ISFOA) was used for the parameter of NTM optimization. Overall, this study designed an intelligent decision-making system with privacy protection, temporal modeling, and adaptive optimization features. The main contributions of this study are summarized as follows:

- This study develops an FL-based distributed framework where a central server coordinates local deep learning model training across manufacturing enterprise clients, achieving accurate financial risk prediction without raw data disclosure.
- An improved starfish optimization algorithm (SFOA) has been designed, which incorporates the reconnaissance and foraging strategies of artificial bee colony (ABC) algorithm into the search strategy of the improved SFOA (ISFOA), aiming to improve the convergence accuracy of SFOA and optimize the hyperparameters of deep learning models.
- A hybrid ISFOA-NTM model was constructed as a local deep learning model for the FL framework. Optimizing the hyperparameter combination of NTM model with LSTM controller using ISFOA algorithm, including learning rate, regularization coefficient, and number of neurons, aims to improve the prediction accuracy of local deep learning model.
- The ISFOA-based NTM-FL (ISFOA-NTM-FL) framework has improved the accuracy and generalization ability of financial risk prediction models. On the test set corresponding to 10 epochs, the performance of ISFOA-NTM-FL was validated. Compared with the baseline NTM-FL model and LSTM model, the prediction accuracy of ISFOA-NTM-FL model increased by 13% and 23%, respectively.

The remaining parts of this study are arranged as follows: Section II focuses on reviewing work related to deep learning and federated learning. Section III introduces the methodology and ISFOA-NTM-FL framework of this study. In Section IV, the proposed framework's test results on two test sets are presented. Finally, Section V provides a conclusion of this study.

II. Literature Review

A. Deep Learning and Neural Turing Machine

Deep learning frameworks can model the complex relationships of a system, accurately capturing causal relationships between inputs and outputs. Due to the above advantages of deep learning frameworks, there have been numerous studies using deep learning frameworks in prediction problems at present. The author in [6] proposed a Bayesian-based deep learning model for complex signal processing problems, which significantly improved the performance of deep learning models. Jawed et al. developed a real-time learner learning style recognition model based on deep learning in the context of animation education [7]. This study utilized learners' raw EEG data for evaluation and successfully achieved accurate recognition of learners' learning styles in the context of animation education by constructing an efficient deep learning architecture, providing new tools and methods for the development of personalized education technology. Kim et al. studied the application of deep learning models in predicting the risk behavior of retail investors [8]. The results indicate that this study demonstrates the potential of deep learning models in identifying high-risk investors, providing an effective risk management tool for financial institutions.

The author in [9] proposed a deep learning-based image processing method for quantifying financial audit risks in the healthcare field. This study aims to develop a system that can automatically identify and quantify risks by combining deep learning techniques with financial audit requirements, while providing intelligent solutions for financial management in the healthcare industry. In response to the real-time pedestrian detection, pose estimation, and tracking problems based on visual sensors, a unified deep learning framework integrating multiple deep learning techniques was designed in [10], aiming to improve the accuracy and efficiency of object detection tasks based on deep learning, and provide technical support for intelligent transportation systems and security monitoring fields. Park et al. proposed an investment portfolio management method based on uncertainty perception, which optimized the decision-making process of the investment portfolio by introducing uncertainty factors and combining them with a risk-sensitive multiagent network [11]. Sattar et al. developed a deep reinforcement learning model that combines reinforcement learning with investment strategies to optimize portfolio management in stock trading [12]. The results showed that the proposed deep learning model can significantly improve the returns and stability of investment portfolios. Biswas et al. developed a dual-output time convolutional network with an attention mechanism for stock price prediction and risk assessment, aiming to enhance the model's ability to capture key information [13]. This study improved the

accuracy of financial market forecasting and risk prediction by introducing attention mechanisms.

NTM is an advanced architecture in deep learning models that integrates DNN with classical computational theory. By introducing external memory mechanisms, it extends the capabilities of traditional neural networks to handle more complex sequential tasks, thereby improving the interpretability of DNN. The author in [14] developed an NTM architecture that can process time series data and generate interpretable predictive results by combining data from physician-pharmacist collaborative clinics. Stogin et al. proposed an NTM model that can be proven stable under finite accuracy and time conditions, demonstrating its robustness and reliability under limited computing resources, providing a foundation for the deployment of neural Turing machines in practical applications [15]. The author in [16] designed an NTM model based on differentiable decision trees and introduced the finite time difference method into NTM, aiming to solve electromagnetic problems. Wu et al. revealed the emergence mechanism of machine thinking in complex systems through theoretical modeling and experimental analysis [17]. Postlethwaite et al. extended the computational power of traditional Turing machines by introducing continuous time dynamics theory, providing a new framework for modeling and optimizing complex systems [18].

B. Federated Learning

Federated learning has the advantage of protecting data privacy. How to train the dataset on local devices and upload the model parameters generated during the training process to the central server is a key issue. Therefore, federal learning is widely used in a large number of financial, pharmaceutical and Internet enterprises. Mistry et al. designed a federated learning framework for privacy-preserving screen activity tracking and classification problems [19]. This study uses a federated learning framework based on an electronic learning background to track and classify screen activity in real-time. The results show that the federated learning framework effectively avoids centralized storage of sensitive data through local data processing and model aggregation. Orlandi et al. explored the problem of federated learning in processing independent data in edge intelligence environments [20]. This study proposes an entropy-based approach that improves the performance of federated learning on edge devices by optimizing data distribution. Xie et al. designed a federated learning framework based on the problem of multi-target recognition that combines differential privacy technology, which can achieve accurate recognition of multiple task targets while protecting data privacy [21].

Kalapaaking et al. combined blockchain technology with federated learning and proposed a secure aggregation framework based on a trusted execution environment, suitable for internet of things (IoT) scenarios [22]. This study ensures data security and traceability through blockchain while utilizing federated learning for distributed model training. The author in [23] provided a comprehensive review of privacy inference attacks and defenses in centralized and federated learning, while also introducing the potential applications of federated learning in privacy-sensitive fields. In [24], the author proposed a federated learning framework that combines server learning and effectively alleviates the challenges posed by non-independent and identically distributed data through global model optimization on the server side. Sun et al. introduced federated learning based on the background of machine fault diagnosis tasks, gradually training the model to learn data features from simple to complex, effectively reducing the impact of noisy labels on model performance [25].

III. Materials and Methods

In order to improve the accuracy of predicting financial risks in manufacturing enterprises and enhance the intelligence level of financial risk management, this study designed a hybrid NTM-FL framework based on FL architecture and NTM model for financial risk prediction. This hybrid framework uses a distributed architecture to protect the privacy data of enterprises, while the NTM model with LSTM controller is trained on each local device. In the financial risk prediction model for manufacturing enterprises based on the NTM-FL framework, the client (participant) is a local device that collects core data of the manufacturing enterprise. The local device uses a deep learning model to train the parameters of the risk prediction system. The central server is responsible for coordinating the training of financial risk prediction models and aggregating updates of local models. During the FL framework training process, data from distributed local devices is always stored on the local devices, and only model parameters are shared. In the FL model prediction process, each local device is trained using NTM models based on data from different manufacturing companies. Fig. 1 shows the designed hybrid NTM-FL framework.

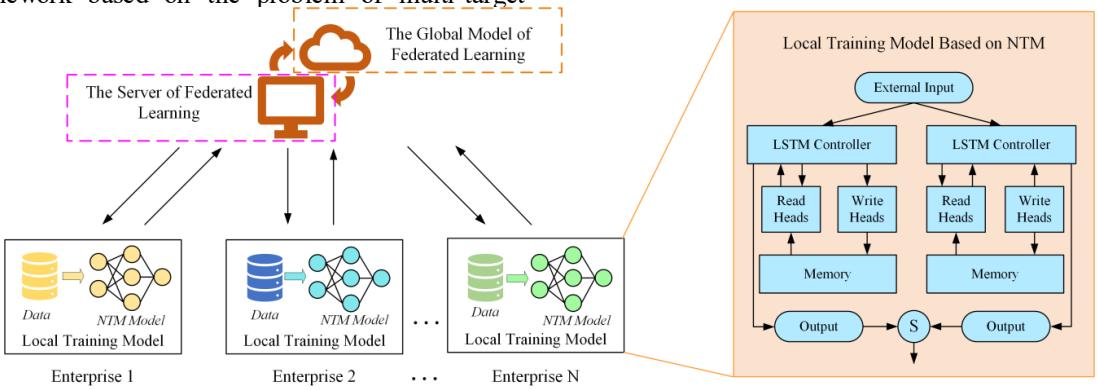


Fig. 1. The hybrid NTM-FL framework for financial risk management in manufacturing enterprises.

A. The Hybrid NTM-FL Approach

In this study, a federated learning framework was used to predict the financial risks of manufacturing companies, aiming to train a global financial risk prediction model for manufacturing companies using time-series data from multiple local devices. Fig. 2 shows the flowchart of the FL framework. Each local device is trained using NTM. Algorithm 1 demonstrates the pseudocode of NTM. In Algorithm 1, $T_{\max}$ is the maximum number of training iterations and k is the current number of training iterations.

Algorithm 1: Neural Turing Machine (NTM)

Initialize: Input layer size, hidden layer size, memory size, number of heads.

NTM.Controller ← LSTM

NTM.Memory ← zeros (memory size)

Training frequency ← T_{max}

Compute: Output layer status.

While (k < Training frequency) do

  Forward propagation

    NTM.Controller processes input.

    NTM.Memory reads memory.

    NTM.Memory writes into memory.

    Generate predictions.

    k ← k+1

  End

Return (Output layer status).

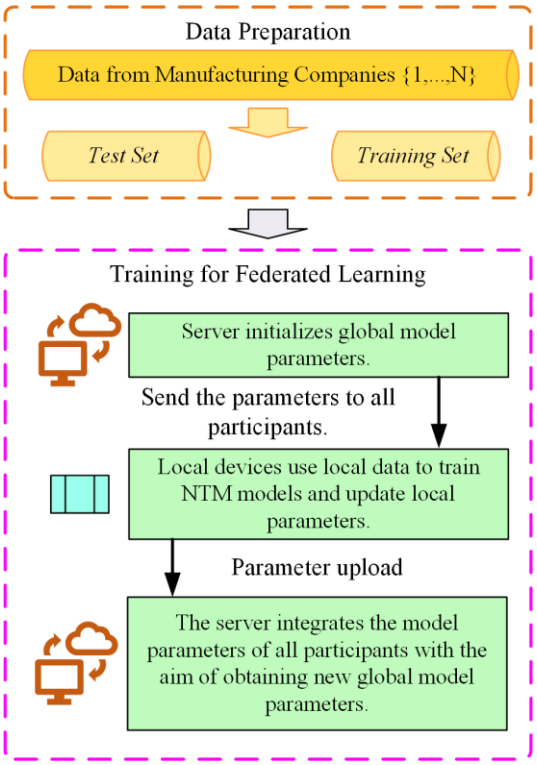


Fig. 2. The flowchart of the FL model.

B. Improved Starfish Optimization Algorithm

This study uses a federated learning architecture to design financial risk management systems for manufacturing companies, aiming to protect their data privacy. At the local device layer, the differential controller of NTM automatically encodes various heterogeneous data, such as financial reports and supply chain logistics costs of manufacturing companies, into temporal feature vectors, and performs causal inference based on the training set in memory. Therefore, in order to further improve the prediction accuracy of the NTM algorithm at the local device layer, this study uses metaheuristic algorithms to optimize hyperparameters such as scaling factors, interpolation weights for new and old content addressing, learning rates, and gradient clipping thresholds. Specifically, in this study, an improved SFOA (ISFOA) algorithm was designed based on the Artificial Bee Colony (ABC) algorithm [26] and the Starfish Optimization Algorithm (SFOA) algorithm [27]. The optimization process of the ISFOA algorithm is shown below.

Step 1. Initialize the population and parameters of the ISFOA algorithm. Generate L solutions based on the maximum and minimum range of hyperparameters in NTM, where each solution $SF_l=[S_l^1,S_l^2,\cdots S_l^X]$ is an X-dimensional vector. Therefore, the population matrix of the ISFOA algorithm is as follows:

$$SF=\begin{bmatrix} SF_1 \\ SF_2 \\ \cdots \\ SF_L \end{bmatrix} \tag{1}$$

Step 2. The exploration stage is based on the starfish optimization mechanism. Generate a random number P_{sa} on the interval [0,1], and for the $x-th$ individual in each solution, if $P_{sa}>0.5$, execute an exploration strategy based on the starfish optimization mechanism. When implementing the exploration strategy in the starfish optimization mechanism, generate a random number $Rand1$ in the interval [-1,1]. If $Rand1>0.5$, update based on the starfish exploration environment strategy, which is defined in (2); Otherwise, update based on the starfish predation strategy, which is defined in (3).

$$S_l^x_new=\begin{cases} S_l^x+(2\times Rand2-1)\pi\times(S_{best}^x-S_l^x)\times\cos\theta, & Rand2<0.5 \\ S_l^x-(2\times Rand2-1)\pi\times(S_{best}^x-S_l^x)\times\sin\theta, & Rand2\geq 0.5 \end{cases} \tag{2}$$

where, $S_l^x_new$ is the parameter of the updated solution vector. S_{best}^x is the solution vector of the currently searched optimal solution. $Rand2$ is a random number on the [0,1] interval.

$$S_l^x_new=S_l^x+Rand3\times(S_{best}^x-S_{rands1}^x)+Rand4\times(S_{best}^x-S_{rands2}^x) \tag{3}$$

where *Rand*3 and *Rand*4 are random numbers on the [0,1] interval, respectively. S^x_{rands1} and S^x_{rands2} are two randomly selected solution vectors from all current solution vectors.

θ is the angle variable for search, which is defined in (4).

$$\theta = \frac{\pi}{2} \times (\alpha / \alpha_{\max}) \tag{4}$$

where α is the current iteration count of the ISFOA algorithm, and $\alpha_{\max}$ is the maximum iteration count of the ISFOA algorithm.

In addition, Fig. 3 illustrates the process of the starfish predation strategy.

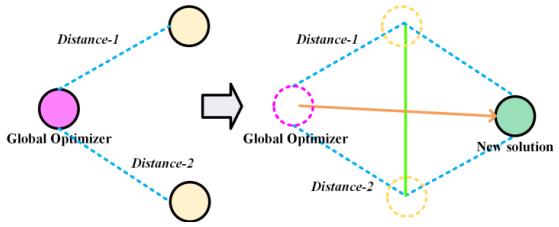


Fig. 3. The predation strategy of ISFOA

Step 3. The development stage is based on the foraging mechanism of bees. If $P_{sa} \le 0.5$, the development phase based on the bee foraging mechanism will be executed. The definition of development strategy based on bee foraging mechanism is as follows:

$$S^x_l_new = \begin{cases} S^x_l + \phi_{lx} \times (S^x_l - S^x_{rands1}), & Rand5 < 0.5 \\ S^x_{\min} + \phi_{lx} \times (S^x_{best} - S^x_{\min}), & Rand5 \ge 0.5 \end{cases} \tag{5}$$

where, *Rand*5 is a random number on the interval [0,1]. $S^x_{\min}$ is the $x-th$ vector of the worst solution so far.

Step 4. Update the global optimal solution. Calculate the fitness function of the current solution of the ISFOA algorithm based on the objective function $f(\hat{y}, y)$, while updating the fitness function of the optimal solution. The definition of $f(\hat{y}, y)$ is as follows:

$$f(\hat{y}, y) = \frac{\sum |\hat{y} - y|}{N} \tag{6}$$

where $\hat{y}$ is the predicted value of financial risk. y is the true value of financial risk. N is the size of the test set.

Step 5. Output the optimal solution. Determine whether the maximum number of iterations has been reached, and if so, output the global optimal solution vector.

IV. Results

This study designed a hybrid NTM-FL framework aimed at predicting financial risks in manufacturing enterprises using both NTM and FL architectures. In addition, to further improve the accuracy of the hybrid NTM-FL prediction framework in financial risk prediction problems, we optimized the hyperparameters of NTM using the ISFOA algorithm. To evaluate the application effectiveness of the ISFOA-NTM-FL model in financial risk prediction problems. Validate based on a dataset consisting of interim and annual reports disclosed by 263 manufacturing companies listed on the Hong Kong Stock Exchange (HKEX) and the New York Stock Exchange (NYSE) from 2015 to 2024. Therefore, when conducting time series forecasting of financial risks, a prediction is made every 6 months, with 20 samples corresponding to each enterprise in the dataset. The sample size of the financial risk prediction dataset is 5260. The dataset includes financial expenditures and profit margins of manufacturing enterprises. Based on the above data, we trained the ISFOA-NTM-FL model for 10 epochs. Therefore, in each epoch experiment, data from 100 samples in the dataset were used to test the ISFOA-NTM-FL model. Before conducting the experiment, the scores, levels, and classifications of financial risks are defined in Table I. The indicators for evaluating the application effectiveness of ISFOA-NTM-FL in 10 epochs are defined in Table II [28]-[30].

TABLE I. The Classifications of Financial Risks

Risk Score	Risk Level	Risk Classification
0. ≤ Score <2	Low Risk	Class 1
2 ≤ Score <4	Low to Medium Risk	Class 2
4 ≤ Score <6	Medium Risk	Class 3
6 ≤ Score <8	Medium to High Risk	Class 4
8 ≤ Score <10	High Risk	Class 5

TABLE II. Evaluation Indicators for Risk Prediction Models

Index	Definition
Macro-Precision	$Macro-Precision = \frac{1}{M}\sum_{m=1}^{M}\frac{TP_m}{TP_m + FP_m}$
Macro-Recall	$Macro-Recall = \frac{1}{M}\sum_{m=1}^{M}\frac{TP_m}{TP_m + FN_m}$
Macro-F1 Score	$Macro-F1\ Score = \frac{2 \times Macro-Precision \times Macro-Recall}{Macro-Precision + Macro-Recall}$

TP_m represents the number of samples in the financial risk prediction test set whose true value is class m and were predicted as class m. FP_m represents the number of samples in the financial risk prediction test set whose true value is not class m but were predicted as class m. FN_m represents the number of samples in the financial risk prediction test set whose true value is class m but were predicted incorrectly.

In addition, the ABC-NTM-FL model, classic NTM-FL model, and classic LSTM model were used to compare with the proposed ISFOA-NTM-FL model, aiming to demonstrate the potential of the proposed ISFOA-NTM-FL model in predicting financial risks in manufacturing enterprises. Fig. 4 shows the financial risk prediction results of the ISFOA-NTM-FL model for two manufacturing companies. As previously mentioned, this study developed hyperparameters for ISFOA optimized NTM model with LSTM controller. In Fig. 5, the loss function curves of ISFOA and ABC algorithm optimized

NTM model with LSTM controller during 15 training processes in epoch 1 are shown.

TABLE III. THE RESULTS OF FOUR LOCAL DEEP LEARNING MODELS

Algorithm	The Parameters of the Optimization Algorithm		Hyperparameters of NTM		Loss Function		
	Population Size	*Number of Iterations*	*Learning Rate*	*Number of Neurons*	*Average Value*	*The Upper Bound of 95% C.I.*	*The Lower Bound of 95% C.I.*
ISFOA-NTM	80	100	0.0093	118	0.0843	0.1301	0.0384
ABC-NTM	80	100	0.0087	124	0.2651	0.3344	0.1958
NTM	-	-	0.0010	124	0.4978	0.5745	0.4212
LSTM	-	-	0.0010	124	0.5139	0.5975	0.4658

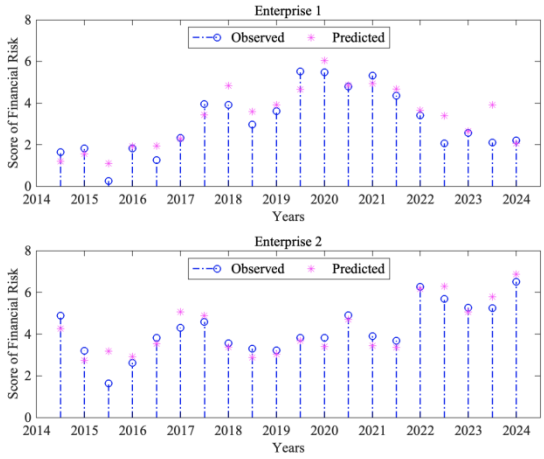


Fig. 4. The ISFOA-NTM-FL model predicts the financial risks of two manufacturing enterprises.

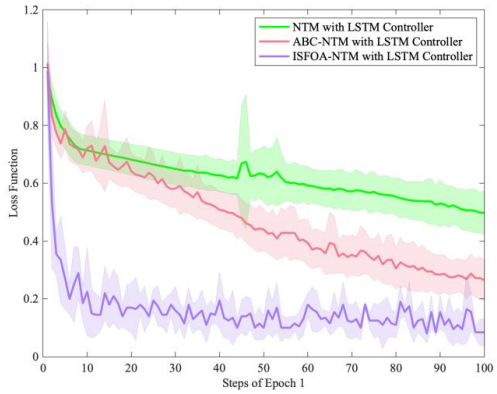


Fig. 5. The loss function curves of ISFOA and ABC algorithm optimized NTM model with LSTM controller.

Table III shows the results of 15 training sessions for four local deep learning models in epoch 1, including ISFOA-NTM, ABC-NTM, NTM, and LSTM. From Table III and Fig. 5, it can be concluded that when optimizing the NTM model with LSTM controller for financial risk prediction, the average loss function value of the ISFOA-NTM model in 15 training sessions is 0.0843, which is lower than other algorithms. Specifically, the average loss functions corresponding to the ABC-NTM, NTM, and LSTM models are 0.2651, 0.4978, and 0.5139, respectively. Compared with the ABC-NTM, NTM, and LSTM models, ISFOA-NTM has reduced losses by approximately 68.19%, 83.07%, and 83.60%, respectively. This result demonstrates the effectiveness of the ISFOA algorithm in NTM hyperparameters, which can effectively improve the accuracy of financial risk prediction models used for manufacturing enterprises.

In addition, the 95% C.I. of ISFOA-NTM is [0.0384, 0.1301], with a width of 0.0917. Compared with the other three models, ISFOA-NTM has the narrowest confidence interval width, indicating that the ISFOA-NTM-FL model has high stability. In addition, the confidence interval of the ISFOA-NTM model has no overlap with other algorithms, and the upper bound of the confidence interval of the ISFOA-NTM model is 0.1301, which is lower than the lower bound of ABC-NTM and significantly lower than the lower bounds of NTM and LSTM. Although the confidence interval of ABC-NTM is better than the benchmark NTM model and benchmark STM model, the confidence interval width is 0.1386, which further reflects that the ISFOA optimization strategy has more advantages in stability. Therefore, ISFOA-NTM achieves a significant reduction in loss function and synchronous improvement in stability through hyperparameter optimization.

When testing the ISFOA-NTM-FL model, ABC-NTM-FL model, NTM-FL model, and LSTM model in Epoch 1, the four indicators of macro-F1 score, macro-precision, macro-recall, and accuracy were used to evaluate the financial risk prediction results. Fig. 6 shows the confusion matrix of the financial risk prediction results of the ISFOA-NTM-FL model in Epoch 1 on a test set consisting of 100 samples. Table IV shows the indicators of the ISFOA-NTM-FL model's financial risk prediction results for five categories in Epoch 1, including the F1 score, precision, recall, and accuracy indicators corresponding to each risk category.

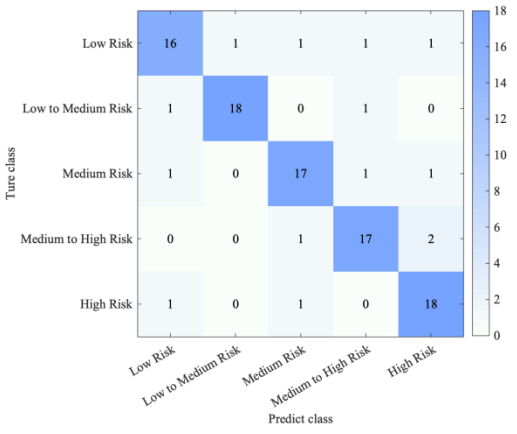


Fig. 6. The confusion matrix of the financial risk prediction results of the ISFOA-NTM-FL model in Epoch 1.

TABLE IV. THE INDICATORS OF ISFOA-NTM-FL MODEL'S FINANCIAL RISK PREDICTION RESULTS FOR FIVE CATEGORIES IN EPOCH 1

Index	Class 1	Class 2	Class 3	Class 4	Class 5
F1 Score	0.8205	0.9231	0.8500	0.8500	0.8571
Precision	0.8421	0.9474	0.8500	0.8500	0.8182
Recall	0.8000	0.9000	0.8500	0.8500	0.9000
Accuracy	0.8000	0.9000	0.8500	0.8500	0.9000

From the data in Table IV, it can be seen that the ISFOA-NTM-FL model has a high overall accuracy in predicting five types of financial risks, with an accuracy of over 80% for each type of financial risk prediction. Among them, the F1 score corresponding to Class 2 is 0.9231, the precision is 0.9474, and the recall rate is 0.9, indicating that the accuracy of risk prediction for this class (Class 2) is the highest. The F1 score for Class 1 is 0.8205, with a recall rate of 0.8000, indicating a 20% real risk of missed detections in Class 1. Overall, the ISFOA-NTM-FL model performs robustly in multiple categories.

Table V shows the training results of four models (ISFOA-NTM-FL model, ABC-NTM-FL model, NTM-FL model, and LSTM model) based on Epoch 1 dataset. Fig. 7 shows the results of the four models corresponding to the four indicators of macro-F1 score, macro-precision, macro-recall, and accuracy. The ISFOA-NTM-FL model achieved optimal performance in three aspects: accuracy, recall, and generalization ability in financial risk prediction tasks. The macro-precision of ISFOA-NTM-FL is 0.8615, and the macro-precision of ABC-NTM-FL is 0.7959. Compared with ABC-NTM-FL, the ISFOA-NTM-FL model has improved macro-precision by 8.24%. The macro-precision of classic NTM-FL and LSTM are 0.7300 and 0.6327, respectively. The ISFOA-NTM-FL model has the highest macro-precision. Compared with the ABC-NTM-FL model, the ISFOA-NTM-FL model can reduce seven prediction errors for every 100 financial risk warnings. The macro-recall of the ISFOA-NTM-FL model is 0.8600, which is an 8.86% increase compared to the ABC-NTM-FL model. The macro-F1 score of the ISFOA-NTM-FL model is 0.8601.

Fig. 8 shows the loss function values of the ISFOA-NTM-FL model over 10 epochs. Fig. 9 shows the corresponding macro-F1 score, macro-prediction, and macro-recall for the financial risk prediction results of the ISFOA-NTM-FL model over 10 epochs. Table VI shows the financial risk prediction results of the ISFOA-NTM-FL model over 10 epochs.

The ISFOA-NTM-FL model exhibits strong generalization ability in 10 epochs. In Epoch 1, the macro accuracy of the ISFOA-NTM-FL model is 0.8615, and the validation loss is 0.0843. In Epoch 10, all metrics of the ISFOA-NTM-FL model achieved the best results, with a macro-precision of 0.9200 and a validation loss of 0.0529. Compared with the results in Epoch 1, the macro-precision in Epoch 10 increased by 6.79%, the macro-recall increased by 5.81%, the validation loss decreased by 37.3%, and the training loss decreased by 47.3%.

The prediction accuracy of the ISFOA-NTM-FL model in handling financial risk prediction problems has improved from 86% in Epoch 1 to 91% in Epoch 10. The average accuracy of the ISFOA-NTM-FL model throughout the entire cycle is 88.5%, which is 2.91% higher than the initial value in Epoch 1. The ISFOA-NTM-FL model maintains high stability with an accuracy of nearly 89% throughout the entire training process.

TABLE V. THE TRAINING RESULTS OF FOUR MODELS BASED ON DATASET OF EPOCH 1

Index	ISFOA-NTM-FL	ABC-NTM-FL	NTM-FL	LSTM
Macro-Precision	0.8615	0.7959	0.7302	0.6327
Macro-Recall	0.8600	0.7900	0.7300	0.6300
Macro-F1 Score	0.8601	0.7901	0.7295	0.6296
Accuracy	0.8600	0.7900	0.7300	0.6300

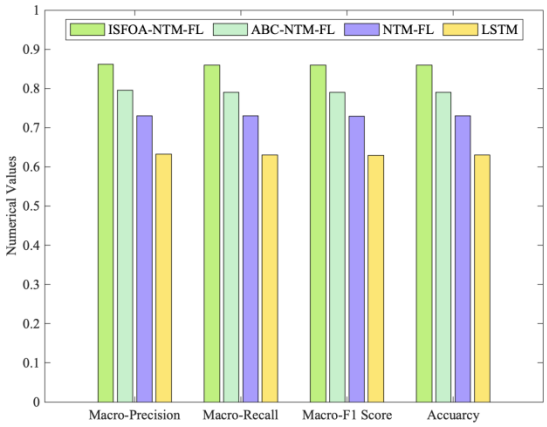


Fig. 7. The results of the four models correspond to the four indicators of macro-F1 score, macro-precision, macro-recall, and accuracy.

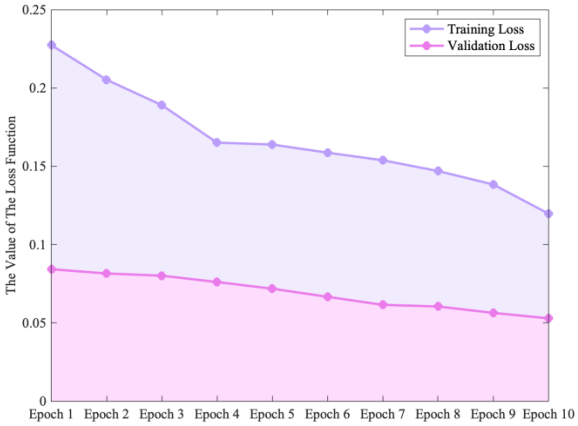


Fig. 8. The loss function of training the ISFOA-NTM-FL model in 10 epochs.

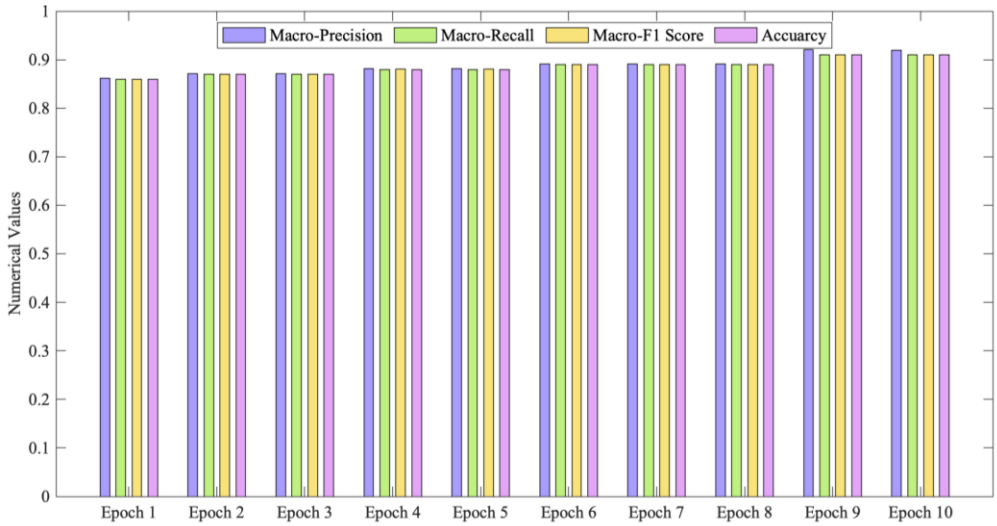


Fig. 9. The results of the ISFOA-NTM-FL correspond to the four indicators of macro-F1 score, macro-precision, macro-recall, and accuracy.

TABLE VI. THE RESULTS OF TRAINING THE ISFOA-NTM-FL MODEL IN 10 EPOCHS

Index	Epoch 1	Epoch 2	Epoch 3	Epoch 4	Epoch 5	Epoch 6	Epoch 7	Epoch 8	Epoch 9	Epoch 10
Training Loss	0.2274	0.2053	0.1891	0.1651	0.1639	0.1587	0.1539	0.1470	0.1384	0.1198
Validation Loss	0.0843	0.0815	0.0801	0.0761	0.0718	0.0667	0.0615	0.0605	0.0564	0.0529
Macro-Precision	0.8615	0.8709	0.8708	0.8812	0.8817	0.8916	0.8915	0.8909	0.9202	0.9200
Macro-Recall	0.8600	0.8700	0.8700	0.8800	0.8800	0.8900	0.8900	0.8900	0.9100	0.9100
Macro-F1 Score	0.8608	0.8705	0.8704	0.8806	0.8808	0.8908	0.8907	0.8904	0.9151	0.9150
Accuracy	0.8600	0.8700	0.8700	0.8800	0.8800	0.8900	0.8900	0.8900	0.9100	0.9100

V. CONCLUSION

In response to the challenges in the accuracy of financial risk prediction for manufacturing enterprises, this study designed a hybrid NTM-FL based intelligent prediction system based on the FL framework and NTM model. On this basis, an ISFOA algorithm is proposed to optimize the hyperparameters of the NTM model with LSTM controller, aiming to improve the prediction accuracy of the NTM model. Overall, the proposed financial risk intelligent prediction system achieves cross enterprise collaborative modeling without sharing raw data through FL architecture. Meanwhile, the system combines the memory enhancement capability of NTM with the temporal modeling advantage of LSTM. The results showed that compared to the ABC-NTM-FL, NTM-FL, and LSTM models, the proposed model improved the accuracy of financial risk prediction by 26.315%. At the end of training in ISFOA-NTM-FL, a prediction accuracy of 91% was achieved. In future research, the application of multimodal data in financial risk prediction models, such as fusion modeling techniques that consider market public opinion, will be further explored.

REFERENCES

[1] Z. Zhou and J. Zhang, "Manufacturing Enterprise Digital Transformation, Financial Flexibility, and Financial Risk-Evidence from China," International Review of Financial Analysis, vol. 104, article no. 104279, 2025. https://doi.org/10.1016/j.irfa.2025.104279

[2] Q. Gao, C. Cheng, and G. Sun, "Big Data Application, Factor Allocation, and Green Innovation in Chinese Manufacturing Enterprises," Technological Forecasting and Social Change, vol. 192, article no. 122567, 2023. https://doi.org/10.1016/j.techfore.2023.122567

[3] T. Teng and L. Ma, "Deep Learning-Based Risk Management of Financial Market in Smart Grid," Computers and Electrical Engineering, vol. 99, article no. 107844, 2022. 10.1016/j.compeleceng.2022.107844

[4] A. Falcon, G. D'Agostino, O. Lanz, G. Brajnik, C. Tasso, and G. Serra, "Neural Turing Machines for the Remaining Useful Life Estimation Problem," Computers in Industry, vol. 143, article no.103762, 2022. 10.1016/j.compind.2022.103762

[5] Tianyu Li and Xiangyu Dai, "Financial Risk Prediction and Management using Machine Learning and Natural Language Processing" International Journal of Advanced Computer Science and Applications (IJACSA), 15(6), 2024. http://dx.doi.org/10.14569/IJACSA.2024.0150623

[6] W. Xu, A. Liu, Y. Zhang, and V. Lau, "Bayesian Deep Learning via Expectation Maximization and Turbo Deep Approximate Message Passing," IEEE Transactions on Signal Processing, vol. 72, pp. 3865-3878, 2024. 3878, 2024, doi: 10.1109/TSP.2024.3442858.

[7] S. Jawed, I. Faye, and A. S. Malik, "Deep Learning-Based Assessment Model for Real-Time Identification of Visual Learners Using Raw EEG," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 32, pp. 378-390, 2024. 10.1109/TNSRE.2024.3351694.

[8] A. Kim, Y. Yang, S. Lessmann, T. Ma, M.-C. Sung, and J. E. V. Johnson, "Can Deep Learning Predict Risky Retail Investors? A Case Study in Financial Risk Behavior Forecasting," European Journal of Operational Research, vol. 283, no. 1, pp. 217–234, 2020. 10.1016/j.ejor.2019.11.007

[9] Y. Ma, "Deep Learning - Based Image Processing for Financial Audit Risk Quantification in Healthcare," Expert Systems, vol. 42, no. 1, article no. e13355. 2025. https://doi.org/10.1111/exsy.13355

[10] Divya Nimma and Arjun Uddagiri, "Advancements in Deep Learning Architectures for Image Recognition and Semantic Segmentation" International Journal of Advanced Computer Science and Applications

(IJACSA), 15(8), 2024. http://dx.doi.org/10.14569/IJACSA.2024.01508114

[11] K. Park, H. -G. Jung, T. -S. Eom, and S. -W. Lee, "Uncertainty-Aware Portfolio Management With Risk-Sensitive Multiagent Network," IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 1, pp. 362-375, 2024. 10.1109/TNNLS.2022.3174642.

[12] A. Sattar, A. Sarwar, S. Gillani, M. Bukhari, S. Rho, and M. Faseeh, "A Novel RMS-Driven Deep Reinforcement Learning for Optimized Portfolio Management in Stock Trading," IEEE Access, vol. 13, pp. 42813-42835, 2025. 10.1109/ACCESS.2025.3546099.

[13] A. K. Biswas, M. S. A. Bhuiyan, M. N. H. Mir, A. Rahman, M. F. Mridha, and M. R. Islam, "A Dual Output Temporal Convolutional Network With Attention Architecture for Stock Price Prediction and Risk Assessment," IEEE Access, vol. 13, pp. 53621-53639, 2025. 10.1109/ACCESS.2025.3551307.

[14] J. Xiao, B. Chen, L. Chen, Q. Wang, S. Tan, H. Yuan, et al., "Interpretable Time-Series Neural Turing Machine for Prognostic Prediction of Patients with Type 2 Diabetes in Physician-Pharmacist Collaborative Clinics," International Journal of Medical Informatics, vol. 195, article no. 105737, 2025. 10.1016/j.ijmedinf.2024.105737

[15] J. Stogin, A. Mali, and C. L. Giles, "A Provably Stable Neural Network Turing Machine with Finite Precision and Time," Information Sciences, vol. 658, article no. 120034, 2024. 10.1016/j.ins.2023.120034

[16] Y. Chen, Y. Zhang, H. Wang, N. Feng, L. Yang, and Z. Huang, "Differentiable-Decision-Tree-Based Neural Turing Machine Model Integrated Into FDTD for Implementing EM Problems," IEEE Transactions on Electromagnetic Compatibility, vol. 65, no. 6, pp. 1579-1586, 2023. 10.1109/TEMC.2023.3273724.

[17] X. Wu, Z. Zheng, and J. Weng, "On Skull-Closed Machine Thinking Based on Emergent Turing Machines," IEEE Transactions on Artificial Intelligence, vol. 5, no. 6, pp. 3057-3071, 2024. 10.1109/TAI.2023.3337322.

[18] C. M. Postlethwaite, P. Ashwin, and M. Egbert, "A Continuous Time Dynamical Turing Machine," IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 4, pp. 6503-6515, 2025. 10.1109/TNNLS.2024.3397995.

[19] D. Mistry, M. F. Mridha, M. Safran, S. Alfarhood, A. K. Saha, and D. Che, "Privacy-Preserving On-Screen Activity Tracking and Classification in E-Learning Using Federated Learning," IEEE Access, vol. 11, pp. 79315-79329, 2023. 10.1109/ACCESS.2023.3299331.

[20] F. C. Orlandi, J. C. S. Dos Anjos, V. R. Q. Leithardt, J. F. De Paz Santana, and C. F. R. Geyer, "Entropy to Mitigate Non-IID Data Problem on Federated Learning for the Edge Intelligence Environment," IEEE Access, vol. 11, pp. 78845-78857, 2023. 10.1109/ACCESS.2023.3298704.

[21] R. Xie, C. Li, X. Zhou, H. Chen, and Z. Dong, "Differentially Private Federated Learning for Multitask Objective Recognition," IEEE Transactions on Industrial Informatics, vol. 20, no. 5, pp. 7269-7281, 2024. 10.1109/TII.2023.3342897.

[22] A. P. Kalapaaking, I. Khalil, M. S. Rahman, M. Atiquzzaman, X. Yi, and M. Almashor, "Blockchain-Based Federated Learning With Secure Aggregation in Trusted Execution Environment for Internet-of-Things," IEEE Transactions on Industrial Informatics, vol. 19, no. 2, pp. 1703-1714, 2023. 10.1109/TII.2022.3170348.

[23] B. Rao, J. Zhang, D. Wu, C. Zhu, X. Sun, and B. Chen, "Privacy Inference Attack and Defense in Centralized and Federated Learning: A Comprehensive Survey," IEEE Transactions on Artificial Intelligence, vol. 6, no. 2, pp. 333-353, 2025. 10.1109/TAI.2024.3363670.

[24] V. S. Mai, R. J. La, and T. Zhang, "A Study of Enhancing Federated Learning on Non-IID Data With Server Learning," IEEE Transactions on Artificial Intelligence, vol. 5, no. 11, pp. 5589-5604, 2024. 10.1109/TAI.2024.3430250.

[25] W. Sun, R. Yan, R. Jin, R. Zhao, and Z. Chen, "Curriculum-Based Federated Learning for Machine Fault Diagnosis With Noisy Labels," IEEE Transactions on Industrial Informatics, vol. 20, no. 12, pp. 13820-13830, 2024. 10.1109/TII.2024.3435449.

[26] S. Lin, F. Li, X. Li, K. Jia, and X. Zhang, "Improved Artificial Bee Colony Algorithm Based on Multi-Strategy Synthesis for UAV Path Planning," IEEE Access, vol. 10, pp. 119269-119282, 2022. 10.1109/ACCESS.2022.3218685.

[27] C. Zhong, G. Li, Z. Meng, H. Li, A. R. Yildiz, and S. Mirjalili, "Starfish Optimization Algorithm (SFOA): a Bio-Inspired Metaheuristic Algorithm for Gglobal Optimization Compared with 100 Optimizers," Neural Computing and Applications, vol. 37, no. 5, pp. 3641–3683, 2025. 10.1007/s00521-024-10694-1

[28] Khan, F. M., Haider, Z. A., Khan, M. O., Ullah, A., Khan, M. S., Fayaz, M., Ahmad, M., & Nabila (2025). Intelligent Cyber-Attack Detection for Autonomous Vehicles Using Advanced Deep Learning Models. ICCK Transactions on Advanced Computing and Systems, 2(2), 27–39. https://doi.org/10.62762/TACS.2025.952297

[29] Alam, N., Zhu, Y., Shao, J., Usman, M., & Fayaz, M. (2025). A Novel Deep Learning Framework for Brain Tumor Classification Using Improved Swin Transformer V2. ICCK Transactions on Advanced Computing and Systems, 2(1), 1–10. https://doi.org/10.62762/TACS.2025.807755

[30] Tiwari, S., Kumar, D., Mahajan, A., & Sachar, S. (2025). Emotion Detection from Speech Using CNN-BiLSTM with Feature Rich Audio Inputs. ICCK Transactions on Machine Intelligence, 1(2), 80–89. https://doi.org/10.62762/TMI.2025.306750

User Requirements of Adaptive Learning Through Digital Game-Based Learning: User-Centered Design Approach to Enhance the Language Literacy Development

Nur Atiqah Zaini[1], Tengku Siti Meriam Tengku Wook[2], Mohd Nor Akmal Khalid[3], Shahrul Azman Mohd Noah[4]
Center for Software Technology and Management[1, 2], Center for Artificial Intelligence Technology[3, 4]
Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Malaysia[1, 2, 3, 4]
Design and Technology Centre (DeTeC), School of Computing and Creative Media,
University of Technology Sarawak, 96000 Sibu Sarawak, Malaysia[1]

***Abstract*—This study aims to elicit the user requirements of digital game-based learning among the primary students through adaptive digital game-based learning, with a focus on enhancing language literacy. Hence, acquiring the user requirements through user-centered design approach is emphasized to identify the specifications and provide practical insights for language learning and digital literacy skills. Further, the requirement specifications are specifically aligned to promote the quality of education for designing the digital game-based learning of language literacy, considering the game elements such as feedback mechanism, player's profile, game rules, game genre, game environment, rewards, adaptive language learning contents, integration of virtual tutor through artificial intelligence, activities and challenges. This paper presents a qualitative analysis of the results from a controlled study that investigates the potential of digital game-based learning through adaptive learning for the enhancement of language proficiency. Thus, it can contribute to the broader field of digital game-based learning by expanding the understanding of adaptive learning for language literacy as an optimal strategy where the primary school students can experience the influence of artificial intelligence technology development nowadays.**

***Keywords*—Digital game-based learning; adaptive learning; artificial intelligence; language literacy; optimization; primary students; user experience**

I. Introduction

Digital game-based learning of language literacy has gained prominence as an innovative approach that leverages engagement and interactivity in adaptive learning systems. The adaptability of learning should maintain the students' engagement based on their needs and how students benefit from the contextual learning through educational games implemented [1], [2], [3]. Additionally, the digital game-based learning offers a dynamic and captivating platform for students in language learning by providing the immersive environments and interactive narratives. Thus, this study examines the rationale for incorporating the optimization of digital game-based learning into language learning by analyzing relevant theories, game mechanics, game elements, adaptive digital game-based learning and artificial intelligence from pertinent studies [1], [4], [5], [6], [7], [8]. Furthermore, acquiring the user specifications will be aligned with the users' needs, fostering the positive impacts on the adaptive learning through edutainment approaches.

To illustrate this contextualization of adaptive learning through digital game-based learning, considering the learning contents are needed which align to the syllabus and features embedded in the existing language learning applications. These features incorporated gamified elements that offer an enjoyable and motivating experience for the learners, specifically the primary school students. They could engage with the learning contents provided through the gaming environment, yet enhancing the language literacy skills [6], [9], [10], [11], [12]. Moreover, the integration of digital game-based learning extends the benefits of adaptive learning by fostering gamified elements and specific language learning contents. In such, digital game-based learning engages the primary school students in terms of interactivity and enjoyable learning experience, where they are able to enhance the cognitive skills but also support the diverse learning styles. Despite the growing attention on digital game-based learning, there is a limited understanding of how adaptive learning strategies can be effectively applied to the language literacy among the primary school students. By emphasizing on the beneficial impacts of educational games, however, the important aspects such as cognitive, linguistic, feedback mechanisms and digital literacy development remain insufficiently addressed.

Recently, there has been increased interest in the potential use of adaptive learning systems due to the transformation of the educational paradigm, particularly during COVID-19, which has enforced a shift to virtual learning systems [13], [14], [15], [16]. The adaptation of technology became a crucial aspect of ensuring continuity in education and the consistency of information deliverance. With the limitations of conventional teaching methods, adaptive learning tools were implemented. These tools enable virtual personalization of the learning experience and communication for students. However, with the surge in digital activities, there were challenges highlighted such as limitations on the deliverance of information, disparity of access to high-speed internet in rural

areas, digital literacy among students and teachers, and adaptivity of virtual learning [17], [18], [19], [20]. Google Classroom, Google Meet, Microsoft Team and other mobile learning applications were focused on, in which these platforms were designed to facilitate remote and hybrid learning environments.

Additionally, over the past few decades, Malaysia, as a developed country, has made substantial investments in education and technology development. To meet the demands of the 21st century, Malaysia has initiated a strong emphasis on Science, Technology, Engineering and Mathematics (STEM) education to increase the students' interest and proficiency starting from primary to tertiary levels [21], [22], [23], [24]. Thus, recognizing the importance of digital literacy among the students, transformation plans and strategies are developed to equip them with digital skills and promote virtual learning. Malaysia has focused on leveraging education and technology while facing challenges in reforming the education system, including the implementation of Artificial Intelligence (AI) to enhance learning tools [25], [26], [27]. The country has prioritized education as a key to success by improving access, quality, equity and efficiency, thereby producing students who are competitive, creative and proficient in both languages, such as Malay and English.

Malay language is the primary language of Malaysia, where it serves as the national and official language used in various sectors such as government, laws, public communication and education too. As the Malay language is being used extensively in schools, all Malaysian students need to be proficient in the language to improve their vocabulary and literature aspects. Despite the dominance of Malay, Malaysia is a multilingual nation in which Malaysians are bilingual or even trilingual [28]. This linguistic diversity is especially pronounced in the state of Sarawak, Malaysia, where multiple languages and dialects are commonly spoken [29]. However, the proficiency of the Malay language remains limited due to the dominant use of local dialects, considering the lack of foundational understanding in that language [24], [30], [31], [32], [33].

Thus, students may not be exposed to standard Malay frequently enough, and this inconsistency can affect their ability to understand and use standard Malay in their daily lives. Therefore, the difficulties in understanding lessons and applying them in daily lives should be considered, where a good command of standard Malay is often required, specifically for primary school students. Limited practice in standard Malay may affect the development of comprehensive language literacy skills, such as reading, writing and speaking [29]. Hence, there is a need to improve the strategies of learning the Malay language through the implementation of technologies to potentially develop strong foundational skills in standard Malay among the primary school students. In fact, it encourages them to communicate more efficiently [34], [35]. Hence, it is beneficial for their future learning, as establishing strong foundational skills in standard Malay contributes to sustained language development and communicative competence. Such proficiency provides a critical basis for advanced literacy acquisition, academic achievement, and lifelong learning.

With the rapid digital transformation of educational institutions, specifically for schools, students can access learning materials at their convenience and learn at their own pace. Hence, this flexibility is particularly beneficial for improving their digital literacy and enhancing their language literacy skills through the integration of digital game-based learning, which is the focus area of this research study. Correspondingly, the digital game-based learning provides various subjects on mobile and other platforms where the primary school students are exposed to a diverse number of learning contents [34], [36], [37]. However, due to the notion and concerns that educational stakeholders, such as teachers, might also not be conversant with the use of digital game-based learning to be conducted in the classroom environment. Therefore, the exposure of 2D and 3D digital game-based learning applications may not deliver the full learning potential due to not focusing on the systematic process of design and development that attracts students in language learning [23], [38], [39], [40], [41]. Thus, digital literacy plays a critical role for teachers, particularly in ensuring that information can be delivered effectively. Research should place equal emphasis on both user interface design and the quality of learning content, rather than prioritizing interface design only.

Numerous studies have discussed the implications of digital game design and development with the use of technologies that can address the concerns of parents, teachers and students. Explicitly, some of the games were designed with various themes of war and vengeance as the main contents that may cause harm for the young learners [6], [42], [43], [44], [45]. Therefore, they might be exposed to the potential distractions of inappropriate language. In fact, the negative implications of game concepts should be avoided as the young learners tend to observe and imitate what they see. Thus, designing and developing the digital game-based learning that emphasizes learning objectives that align with the curriculum would serve as the major source of learning. Contrary to gamification, digital game-based learning focuses on a particular learning skill that makes use of the gaming environment and game design for a specific learning purpose [46], [47]. It emphasizes the development of specific learning skills by focusing on learners within a structured gaming environment that aligns educational objectives with gameplay. Through the deliberate use of game design, digital game-based learning ensures that learning outcomes are explicitly defined and systematically assessable.

Meanwhile, digital game-based learning employs game mechanics and game dynamics that enable learning to become more interactive and presentable, thereby enhancing language proficiency. However, the limitation of feedback mechanisms in digital game-based learning development has become one of the major issues in the research study. The classification of feedback mechanisms for designing and developing has remained unaddressed. Based on psychological perspectives, designing the games involves game elements, such as feedback mechanisms which relate to the students' overall development and psychological experiences [48], [49], [50], [51], [52]. The growing interest in research on feedback mechanisms is becoming increasingly ubiquitous, thus influencing the gamified experience among primary school students in the

field of adaptive learning systems through the integration of gamified learning environments [45], [53]. There is growing research interest in how feedback mechanisms apply in learning, showing just how important they are for students' experiences. For primary school learners, feedback has become an important aspect of gamified environments, especially when used to support adaptive learning.

Thus, integrating the digital game-based learning into the school curriculum is crucial for addressing the challenges and issues in the context of language learning among the primary school students. The learning technique specified on the use of games embraces the design of interactivity that progressively teaches the concept to achieve the learning objectives [6], [31], [48], [80], [81], [82]. Games often incorporate elements such as rewards, challenges and interactivity to encourage students to participate actively. Therefore, enhancing the cognitive skills among the students often requires active problem-solving and decision-making, thus influencing a better understanding of learning, specifically on language learning [54]. Through learning milestones or game levels provided as the game goals might help the students to increase their motivation and engagement in the educational field [40]. Besides, engaging them through educational games offers a range of benefits that can enhance the educational experience, adaptive learning and develop digital literacy skills by using technology effectively [42], [55], [56], [57], [83], [84]. Learning milestones and game levels embedded in the game development can help to boost students' motivation and keep them engaged in their studies. The implementation of educational games in learning strategies also bring wider benefits by supporting adaptive learning and building digital literacy skills through technology use among the learners.

This research study aims to examine the potential of digital game-based learning through adaptive learning systems among the primary students by acquiring the user specifications for enhancing their language literacy skills. All of the specifications are emphasized to improve adaptive learning in educational systems and the impact of digital game-based learning that influences the gamified experience in the field of digital learning technologies. Hence, the educational outcomes of digital game-based learning, particularly in the context of language learning, would be investigated in detail through the method of user-centered design.

The structure of this paper is designed through a systematic investigation of digital game-based learning for enhancing language literacy in primary students. Based on the research problems obtained, there is a need to explore a deep understanding of the related issues of digital literacy and language literacy among the primary students. Then, the literature review first examines existing studies on adaptive learning and digital game-based learning as the research focus. The methodology section describes the qualitative methods and user-centered design process used to define specific user requirements. Results and analysis present data from a controlled study which evaluates on how adaptive game features influence the language skills. The discussion section reflects on these findings, connecting them to broader educational technology trends. The conclusion summarizes the implications for classroom integration using technology and outlines future research opportunities in adaptive learning.

II. Related Studies

A. Digital Game-Based Learning

Digital game-based learning has been widely explored as a learning method to enhance the students' engagement and improve their learning outcomes in language literacy. Previous studies suggested that this method could effectively integrate the cognitive and emotional aspects of the students through the combination of traditional instructional methods with the implementation of interactive game elements [58], [59]. By focusing on the game mechanics, such as missions or challenges, game activities, rewards, and progress tracking have been implemented to motivate students and maintain their interest in the educational process and learning content. Conversely, conventional methods of learning remain relevant to be applied to the learning style of the students, but the dynamics of learning have evolved due to technological advancement nowadays [44]. Therefore, the revolution of technologies enabled the digital game-based learning to be designed based on the students' needs and preferences for educational purposes. Thus, digital game-based learning could foster collaborative learning, critical thinking and problem-solving skills by providing them opportunities to learn in diverse and engaging strategies for language literacy skills.

Additionally, the empirical studies have demonstrated that digital game-based learning could significantly improve the learning outcomes in language literacy skills. For instance, an experimental analysis found that students who integrated educational games had higher levels of engagement and showed better performance compared to focusing only on conventional ways of learning [60], [61], [62]. In fact, digital game-based learning has been effectively enhancing the vocabulary, reading comprehension and grammar skills, specifically in mastering the language literacy skills among the students.

Despite the benefits, there are challenges and rising concerns in the research studies regarding the implementation of digital game-based learning in language literacy. Considering these few factors of challenges, such as poor interface design, difficulty in balancing the educational contents with engaging gameplay, misalignment of learning contents with curriculum, varying levels of digital literacy skills among students and teachers [6], [48], [53]. Addressing these challenges requires effective strategies to be developed that optimize the educational value of integrating digital game-based learning while ensuring the accessibility of technologies. Effective strategies should emphasize the integration of adaptive learning to personalize the learning needs, enhance their user experience while interacting with educational games and manage the technology in alignment with the dynamics of learning [42].

Adaptive learning systems can dynamically adjust the learning contents, students' learning pace and complexity of lessons based on the students' progress and preferences. Hence, they can experience personalized learning through the customization of educational experience in adaptive digital

game-based learning. The development of adaptive learning systems is designed to continuously evaluate the students' performance, offering them the benefits of refined contextual language contents and exercises based on their strengths and needs [42], [57], [63], [64], [65], [66]. Innovative learning strategies encourage students to move beyond conventional methods and engage with more diverse approaches to improve knowledge and create more learning styles.

B. Adaptive Learning System

Consequently, the optimization in adaptive digital game-based learning to create an effective gamified learning environment for each individual is important, as it dynamically adjusts the learning contents and instructional approach in response to the personalized learning. Understanding the diverse learning preferences of students plays a crucial role in designing the educational framework and game development, specifically in language learning aspects. There is a need in identifying the students' preferences to enhance the learning experience and overall efficiency of performances in personalized and gamified learning environments [63]. Therefore, the needs of the player's profile in the game development as the character customization was emphasized for the engagement and beneficial impacts on the educational games development [67], [68]. The important aspects of the player's profile as the representative of the player in the virtual game environment may help students feel connected to the game. Thus, it can increase motivation, sustain attention, and encourage active participation throughout the learning process.

On the other hand, the digital game-based learning offers the feedback mechanisms that provide students with immediate responses. As the interaction is needed to interact with the system developed, the needs and preferences of students are highlighted in the research study. Research on digital game-based learning consistently stated that the feedback mechanisms improved the language literacy skills among students [6], [7], [48], [53]. Studies showed that the feedback mechanisms within the digital environment increased the motivation and engagement during the gameplay. By providing them the immediate responses to students' interaction on the system, either positive reinforcement or corrective cues enabled them to be consistently and actively involved in the learning process through the gamified learning environment [48]. This continuous engagement was beneficial in mastering the language proficiency, in which the consistent practice and interaction were particularly improving their social skills development too.

However, the efficiency of feedback mechanisms of explicit, implicit and social feedback remains unaddressed [61], [69], [70]. As the implicit feedback provided the procedure of the digital game-based learning developed, students were able to interact well with the system. Yet, they were provided with clear instructions through texts or audio as the guidance on the gameplay. Meanwhile, the explicit feedback particularly informed the students on right or wrong actions, often accompanied by explanations or instructions to improve their performances during learning sessions through the digital game-based learning. Alternatively, some systems did not provide the feature of feedback mechanism for incorrect actions, in which leaving the users without necessary justifications or explanations that they might need to understand and learn from their mistakes, specifically for improving their language literacy skills [48], [71]. Feedback is crucial in digital learning games, particularly for effective language learning. The effectiveness of explicit, implicit, and social feedback within digital game-based learning can be systematically integrated to enhance the language learning skills of the students.

Several studies have emphasized the importance of integrating the effective feedback mechanisms in digital game-based learning to improve students' learning outcomes and understand their mistakes during learning sessions. If the students did not receive adequate feedback mechanisms during the gameplay, they tended to stop playing the games [53], [72]. It happened due to boredom and losing interests where they would not engage in a dull learning environment through the digital game-based learning developed. In fact, they would suspend from using the digital game-based learning entirely, which would affect their user experiences in the gamified learning environment [48], [53], [69]. Thus, developing the feedback mechanisms that align well with game mechanics and educational learning objectives are technically challenging especially for the game designers that must work within the constraints and limitations to achieve the edutainment goals [49]. Insufficient feedback mechanism integrated in the game design and development may affect the engagement of learning and reduce the enjoyment in the gameplay. In fact, both game mechanics and learning objectives should be aligned with the needs, preferences, and abilities of the students to ensure that the gamified learning environment can be conducted effectively.

Moreover, the adaptive learning platforms rely heavily on the technology of Artificial Intelligence (AI) and machine learning to assess the students' performances during learning sessions [64], [73], [74], [75], [76]. The development of the system enables users to track user inputs, for instance, the quiz results, time spent on tasks to be completed, types of mistakes made to be detected, and adjust the difficulty of learning for the questions or activities conducted. Therefore, with the implementation of Artificial Intelligence in educational activities, a virtual tutor system is designed to provide personalized educational support to the students [64], [77]. It serves as a human tutor, as an adaptive learning system can be enhanced through the integration of AI in the gaming environment to expose the students to various learning experiences of language literacy.

Besides, the Natural Language Processing (NLP) of a virtual tutor allows students to understand the learning process, provide responses to them through questions and feedback. With that, it creates engaging interactions between the students and the system of virtual tutor integrated. While AI virtual tutor as an avatar representation can address the challenges faced by the students who are shy or introverted by providing personalized learning with instant feedback, the lack of direct human interaction becomes a significant limitation for them to interact well [64], [67], [68]. This absence may impact the development of social and communication skills, reducing their motivation due to the lack of two-ways communication

between teacher and students in the conventional learning environments.

III. METHODOLOGY

This research study consisted of 30 primary school students from Sarawak, Malaysia and aged between 7 and 9 years old for the user-centered design activities conducted. By integrating a user-centered design approach ensures the design is guided by learners' actual needs and preferences, which may contribute to these factors of engagement, learning relevance, and the usability of the educational games. The first activity involved all of the students to test the three existing digital game-based learning, such as mobile applications of the Malay language developed for enhancing the language proficiency through observation. Meanwhile, the second activity involved the ideation in sketching the digital game-based learning based on their preferences in terms of learning style and their needs. All of the students were Information Technology (IT) literate and familiar with the use of tablets and smartphones. Therefore, informed consents were obtained as the participants involved are below 18 years old.

Next, the third activity consisted of six teachers that were involved in this experiment to obtain their perspectives on the syllabus contents of the Malay language subject, as the learning contents needed to be simplified in the digital game-based learning development. To obtain the requirements, interview sessions had been conducted among the teachers who were teaching the Malay language subject at primary school. All of them were well-versed in pedagogy, with teaching experience ranging from 5 to 20 years. As they have expertise in implementing various teaching strategies, they were also skilled in both conventional and modern educational practices. All of the experiments conducted are based on the user-centered design method to acquire the user specifications in the research study as shown in Fig. 1 [78].

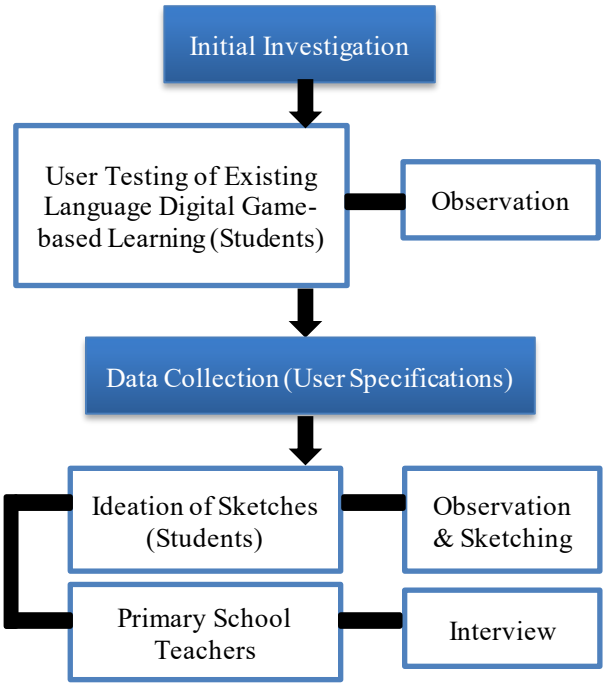


Fig. 1. Process of obtaining user specifications.

A. Experiment 1: User Testing the Existing Digital Game-Based Learning of Language Literacy

This activity involved the three existing digital game-based learning applications of Malay language learning, namely, 1) Mari Belajar Bersama Arif, 2) Belajar Menulis ABC, Suku Kata, and 3) My Latihan – BM T1: Suku Kata I. All of the students were given 30 minutes to explore all of the existing mobile applications by using tablets. They were free to ask the questions regarding activities conducted in the gameplay. The students were required to play and complete the tasks given and experience on using these existing mobile applications.

The evaluation session was carried out based on an observation technique to obtain their responses. Next, the four main highlighted aspects were observed that consisted of feedback mechanisms, game design, user experience and game elements of existing digital game-based learning. The highlighted elements were on the items of character design, use of colour, gameplay, learning contents, game environment, game genre, responses towards the game developed, the needs of a virtual tutor and rewards. Checklist was used as the main instrument to record their preferences and expectations while exploring the existing mobile applications. However, to ensure that unbiased data were obtained throughout the experiment conducted, the students were free to state their preferences, which were not listed in the checklist provided.

The procedure was carried out as follows:

- The respondents were briefed about the objectives of the experiment conducted.
- The respondents were briefed about the three existing digital game-based learning applications of Malay language.
- The respondents were briefed about the items on the checklist form to acquire the needs of them.
- The respondents were given 30 minutes to explore and play with the digital game-based learning applications individually. All of their actions will be recorded.
- The respondents were given the tasks to be completed in each game.
- Throughout the gameplay activities, the respondents would be observed on their digital literacy, Malay language literacy and their needs. They could also give their opinions related to the game design, gameplay and strategies to accomplish the mission for obtaining the rewards.
- Finally, the respondents were given a 10-minutes break before conducting another brainstorming session to sketch their ideas related to the preferences of digital game-based learning.

B. Experiment 2: Ideation of Sketching the Digital Game-Based Learning for Language Literacy

This activity involved all of the students to sketch their ideas in terms of player profile, game environment, game genre, and colour theory that are specified to acquire the user requirements. These requirements would be emphasized in

developing the digital game-based learning for language proficiency. All of the students were tested by using three existing digital game-based learning experiences on the gameplay. Therefore, this experiment was also identifying better understanding among them in mastering the language literacy skills. An observation technique is conducted to evaluate the drawings by the students.

The procedure was carried out as follows:

- The respondents were provided with visual examples to apply in their drawings based on the given instructions.
- The respondents were provided with the wireframe template to draw.
- The respondents are given 30 minutes to draw five main components that they prefer to be included in the digital game-based learning development. These components consisted of character designs (player character, non-player character of a teacher), game environment, gameplay, and colour theory.
- The respondents might draw and colour the drawings based on their preferences.
- The respondents were allowed to ask freely about their drawings.
- Throughout the drawing activities, the respondents would be observed on their preferences to develop the digital game-based learning user interface design through the checklist provided.

C. *Experiment 3: Interview Session with Primary School Teachers to Obtain the User's Needs for Language Literacy*

This activity involved six primary school students who have teaching experience ranging from 5-20 years. The teachers were interviewed on the basis of language learning, learning techniques, integration of technology, students' learning characteristics, feedback mechanisms, language dialect among Sarawakians, digital literacy and language literacy.

The procedure was carried out as follows:

- The interview session would be conducted within 30 - 40 minutes either in person or online.
- The respondents were answering the questions based on their teaching experience in primary school.
- The respondents were asked on the related matters of language literacy, digital literacy, the integration of technology, feedback mechanisms, dominant use of local dialect, impacts of lack of proficiency in mastering Malay language and strategies to enhance language learning.
- All of the interview sessions were being recorded to identify the problems among primary school students in Sarawak, Malaysia.

Hence, all of the experiments were conducted to identify the specific user requirements for designing and developing the digital game-based learning for the primary students. Through observation, interviews and checklists would acquire the needs of the students in identifying the gaps, challenges, approaches and strategies to improve the language literacy skills among them.

IV. Findings and Results

A. *Experiment 1: Analysis and Discussion (Quantitative Data)*

Fig. 2 shows the analysis of quantitative data gathered from the checklists of the experiment. There are 13 items listed in the checklists to identify the usability problems when using the three existing mobile game applications for language learning. There are four categories highlighted when the experiment is carried out. The categories are as follows: 1) game elements, 2) user experience, 3) feedback mechanisms, and 4) game design. The collected data will be analyzed and presented as percentage-based graphs to provide clear visual representation of comparisons of the findings.

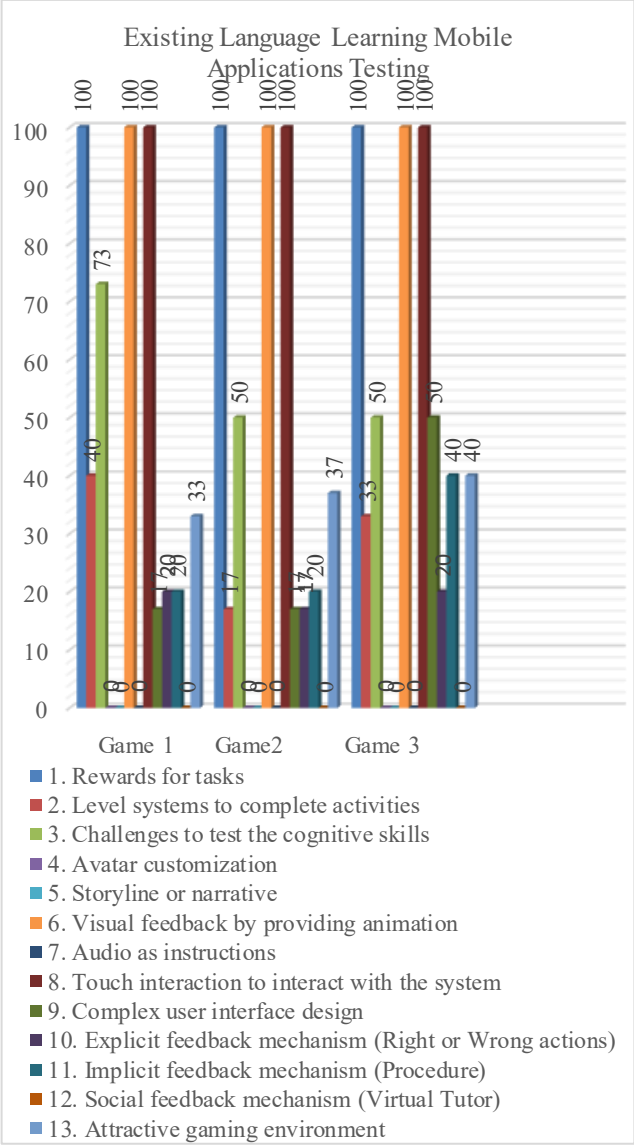


Fig. 2. Comparisons of existing language learning mobile applications.

Based on Fig. 1, the analysis data for Game 1, 2, and 3 show the game elements which are rewards, level systems, challenges, avatar customization, storyline or narrative, audio, and gaming environment needed in the language learning mobile game applications for the primary students. They are exposed to learning the digitized version of the Malay learning language applications through the use of tablets. However, some of the level systems to complete activities are not provided in the game design and development, but only provide them with different tasks to be accomplished. The purpose of level systems enables students to advance through the levels as they complete all the activities from the game itself, where they can unlock the new challenges based on their progress. While the challenges to test cognitive skills are included in the game design, they can be aligned to different levels of knowledge. So, the students may improve their language literacy skills from the lower level to the higher level.

Besides, all of the games do not provide the avatar customization as the profile player in the game design where it shows 0%. The implementation of player profiles through personalized avatars aims to enhance player engagement and improve the interaction with the game where currently the students are well-exposed the avatar customization. Not only that, storyline or narrative shows 0% too in which, the element is essential in game development where it serves to enhance the cognitive skills within educational games. However, these existing mobile game applications do not integrate storyline in guiding the students about the game flow. In fact, the students have a variety of knowledge levels and different skills, where some of them may have difficulties in reading and understanding. Hence, the use of audio to give instructions to the students about the game flow is needed too, where all of the games do not provide any of it.

As the main focus of the study is the feedback mechanisms that are categorized into implicit, explicit, social and visual feedback. Visual feedback, such as animation is provided in the game design. Meanwhile, the implicit, explicit and social feedback indicate the lower percentage when the students are operating all of the games. The inadequate feedback mechanisms may lead them to suspend the learning process of language literacy during the gameplay. It may occur when the absence of timely and effective responses hinders the students' progress, unlike the more responsive interaction provided in the classroom environments. Therefore, complex user interface design may affect the interaction of the students because they do not understand how to navigate and accomplish the tasks efficiently. Based on the analysis, it shows that 17% of the students are having difficulties in understanding the complex user interface design. Misinterpretation of non-clickable images as buttons can result in user frustration, where some of them are just clicking all the images randomly. This may occur due to unclear differentiation between actionable and non-actionable components in the user interface design. Not only that, this problem of usability happens due to their cognitive abilities being limited compared to adults.

B. Experiment 1: Analysis and Discussion (Qualitative Data)

After conducting the session of Experiment 1, Table I shows the analysis of students' preferences for the four main highlighted aspects of existing digital game-based learning of the Malay language, considering the feedback mechanisms, game design, user experience and game elements to be specified in detail through observation technique. The analysis will be conducted by using thematic analysis.

TABLE I. ANALYSIS OF OBSERVATION

Respondents (R)	Remarks
R1 – R30	Respondents prefer the learning environments to be designed in 2D and 3D such as jungle, zoo, outer space and flower gardens. They are also expressing interest in customizing the player profiles as the game character.
R7 – R25	Respondents prefer an active game genre such as adventure game with different levels of difficulties to ensure they engage and enjoy the learning session.
R1 – R30	The game shall not take up a lot of space or storage in the device such as tablets or smartphones. Additionally, the respondents suggest no advertisement showing up while playing which may lead to uninstalling the application and distraction.
R1 – R20	Respondents did not understand the gameplay of the existing mobile applications where they tend to stop playing the games. They are requesting an English version where it is more fun and easier to understand rather than using Malay language. There is no instruction of audio or verbal provided to them, but only focusing on reading the descriptions on how to play only.
R1 – R30	Respondents express themselves that they do not like to read books because of static images, where there is no interactivity at all. Thus, they tend to be bored in reading and learning Malay language.
R1 – R30	The needs of non-player character (NPC) shall be created and developed in the game. So, the avatar will be the representative of a virtual tutor or teacher to help in monitoring their progress track. Due to no feedback interaction between teacher and students in the existing game, lack of active engagement in the learning environment will occur.

Most of the digital game-based learning focuses primarily on visual and audio elements without aligning with the learning objectives and satisfying the students' needs. Given the varying preferences of students, it can be challenging to determine which aspects of the game design should be emphasized. During the experiment conducted, students found the user interface design of digital game-based learning confusing, struggling to understand the functionalities of the buttons in the game layout. This confusion leads them to press the icon images randomly, limiting the usability and reducing their engagement in learning the Malay language through the digital game-based learning.

The inclusion of game elements such as rewards, game instructions, challenges or activities, multimedia elements, avatars, non-player characters for teachers, player profiles, and gameplay can provide a beneficial learning experience in a gamified environment, enhancing the digital and language literacy. However, as the experiment progressed, it became apparent that the students were unable to understand the game learning contents due to a lack of instructional design in the game development. The students tend to stop playing, making it impossible to track their progress in the gameplay which leads to frustration and boredom to explore the game efficiently. Hence, there is a need to emphasize on the learning

contents that align to the learning objectives of the syllabus for improving the Malay language.

Therefore, feedback mechanisms are mainly focused in the research study, as every action taken by the students was recorded to highlight the limitation of the existing digital game-based learning developed. Lack of feedback mechanisms in the game development led to decreased concentration among students during the learning session. They also keep repeating the same mistakes in the gameplay. Additionally, lack of feedback mechanisms can lead to confusion about the flow of the game, thereby the students would lose interest in continuing to play the games. The students do not understand the reasons for their mistakes because they are still in the learning process with cognitive skills that are less developed compared to adults. Hence, there is a need to highlight the feedback mechanisms that can enhance their learning performance in mastering the Malay language proficiency.

Meanwhile, some of the students suggest having a non-player character (NPC) that serves as a teacher in the digital game-based learning environment. They believe this feature will help in monitoring the students' progress and provide instant feedback, as the NPC can answer questions during the learning process. Implementing such an NPC can potentially enhance student engagement by replicating real-life classroom interactions in a virtual learning environment. The integration of NPC in language learning sessions allows students to focus more by offering the interactive functionalities that respond to their questions and provide explanations on various learning topics such as reading, writing and speaking. For their own player's character, the students suggest the customization of avatar where they can design based on their preferences.

As a result, the analysis shows the critical need for enhancing the feedback mechanism, user interface design, functionalities, and learning content related to language literacy within a digital game-based learning environment. These improvements will be beneficial for students to improve their digital and language literacy by following the preferences of their learning style and addressing their specific needs.

C. Experiment 2: Analysis and Discussion (Idea Sketching)

After conducting the experiment, the data collection will be emphasized on the students' drawings based on four primarily main components. Fig. 3, 4, 5 and 6 show the analysis of drawings created by the students through sketching their ideas based on their preferences for digital game-based learning design and development.

Based on Fig. 3 indicates that students prefer to represent themselves as human-form characters in the digital game-based learning development. They can customize the details of these human avatars to play as their in-game representative. Meanwhile, the non-player character representing the teacher can also be developed as a human-form character, as one of the user requirements in the gameplay. Fig. 4 emphasizes the design of various environments, including the zoo, jungle, space and underwater. These environments can be customized as their background theme according to the students' learning preferences, allowing for a more personalized and engaging experience in the virtual world.

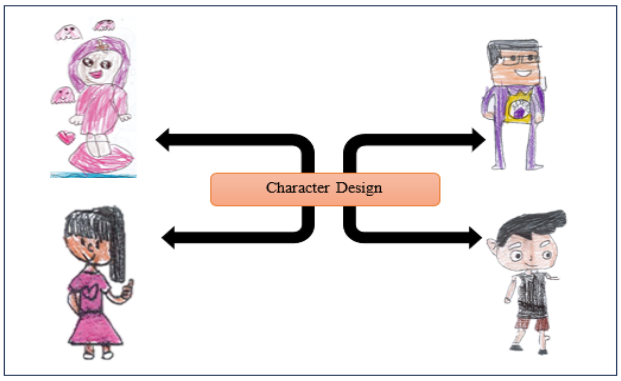


Fig. 3. Character design.

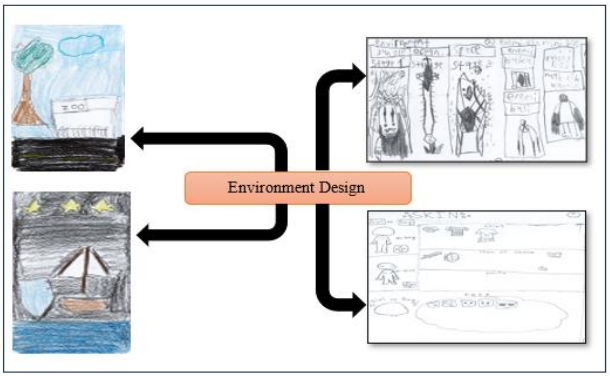


Fig. 4. Environment design.

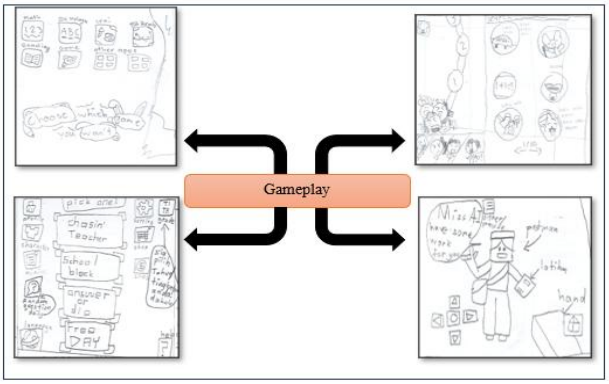


Fig. 5. Gameplay.

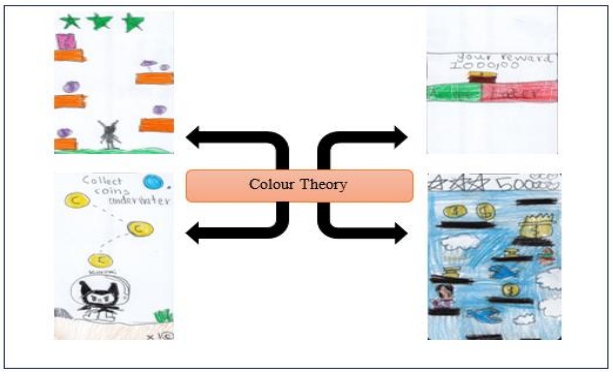


Fig. 6. Colour theory.

Fig. 5 highlights the various elements of gameplay in terms of interaction, user interface design, virtual tutor of artificial intelligence, game mechanics, game objectives, activities, challenges and learning context. This figure provides a comprehensive overview of how these elements may contribute to the overall gaming and learning experience. The ideation sketches focus on the game levels, mini games to be explored by the students in mastering the language proficiency. These game levels provide a structured way to increase the difficulties in the gameplay and maintain the engagement to ensure the players are continually challenged. Through the game levels provide the assessments to help them understand their strengths and areas of improvement in language learning. Therefore, feedback mechanisms as the main focus elements such as explicit, implicit, visual and social feedback will be emphasized based on their needs. Meanwhile, the instant and haptic feedback will be added as complementary features. These mechanisms aim to enhance a better understanding and proficiency in system interaction and digital literacy among the students.

Through the application of colour theory in Fig. 6, every factor of the digital game-based learning design will utilize the colour strategically to capture students' attention and create an engaging learning environment. This use of colour will not only attract them but also support their learning by enhancing the visual appeal. Thus, the learning contents developed will be more accessible and enjoyable as affecting their user experience by the integration of technology. Moreover, the use of bright and vibrant colours can be used to signify progress and rewards for the students' achievements or levels completed. This might boost the motivation and create a sense of accomplishment among the students to engage in learning language by using digital game-based learning.

D. Experiment 3: Analysis and Discussion (Interview Session)

The data collected through the interview session will be presented in thematic analysis as shown in Table II.

TABLE II. ANALYSIS OF INTERVIEW SESSIONS (TEACHERS)

Respondents (R)	Remarks
R1	Respondent 1 states the primary reason for lack of proficiency among students to master the language literacy is due to inadequate foundational knowledge in the Malay language. The dialectal variations significantly impact the language proficiency in terms of speaking, writing and reading. Therefore, the importance of technology implementation, as the learning process should align with technological advancements. However, when educational game applications fail to provide adequate feedback mechanisms during learning sessions, it can lead to challenges such as maintaining the students' engagement and managing their behaviour due to the absence of effective two-ways communication between students and teachers. Given that students are still in a formative stage of cognitive and social development, consistent interaction and structured supervision are crucial as the approaches provide them with proper guidance and support. Although educational games have the potential to stimulate their interest and motivation in enhancing learning, it is important to highlight the clear boundaries on the use of technology. This is due to the risk of gadget dependency, which could negatively impact the students' attention spans and social interactions in the real world. Consequently, these applications are beneficial in enhancing the learning outcomes, the use of the tools must be considered to be implemented. To ensure the complement, rather than replacing the conventional pedagogical methods, this approach will provide a holistic educational experience that nurtures both academic and personal growth in students.
R2	Respondent 2 adopts a diverse array of technology that can enhance the teaching strategies, including the utilization of computer labs and interactive educational game applications in such Kahoot and Wordwall. Relying exclusively only on a single conventional teaching approach can lead to short-span of attention and boredom among the students, thereby their focus on learning will be easily shifted. As the integration of Artificial Intelligence (AI) technology has begun to be introduced to the teachers, whereby the in-house training courses are also being conducted, these may help in improving the teaching and learning efficiently. Correspondingly, R2 states that by incorporating AI technology into educational game applications can help to ease the teachers' workload while also allowing students to be monitored as well. Therefore, by integrating innovative teaching methods with the implementation of technology, the learning contents and game design should align and be relevant with the educational objectives for the students. These methods are crucial as the technological tools are effectively utilized to enhance the teaching and learning process, ultimately promoting better academic performance among students.
R3	Respondent 3 prefers to implement the syllable method in teaching strategies due to its ease of memorization and enhancing pronunciation accuracy. This method is not only supporting vocabulary proficiency but also providing a strong basis for language development. Therefore, the integration of phonics is also beneficial to help students in developing the language literacy skills in alignment with primary education standards. However, the use of the syllable method is primarily recommended for developing the game. R3 emphasizes that students are increasingly attracted to use technology for learning tools, in which they are familiar with current technological advancements. However, it is important that clearer and explicit instructions should be provided before initiating the digital exercises. R3 indicates that verbal instructions are more effective than written ones, as they enhance the clarity of interaction between teachers and students. This may improve communication effectively with an engaging learning environment, which allows for dynamic interaction and active student participation during class sessions. Consequently, by implementing the feedback mechanisms will not only focus on students' engagement, but also contribute to better comprehension and educational outcomes. It proves that an effective instructional method as an approach to teaching, aligns well with educational practices.
R4	Respondent 4 emphasizes that educational game applications have the potential to significantly engage the students through interactive learning in the virtual gaming environment. This approach ensures that the time spent is more beneficial compared to merely playing regular games only for entertainment. Students preferred to use characters of Avatars or specifically implement the player profiles, as these existing game applications allow them to choose their preferences and

Respondents (R)	Remarks
	personality styles. Therefore, the integration of feedback mechanisms in educational games enable the students to respond well with the system developed. These feedback mechanisms are crucial for the game development as needed by the students to understand the game procedure, learning contents and responses towards the system too. By implementing the interactivity and responsive feedback mechanisms enable to maintain the students' attention and reinforce the learning outcomes. If these feedback mechanisms are not adequately integrated, students may less engage and not focus on learning, as the feedback mechanisms are supposed to be mirroring the interactive and responsive nature of traditional classroom environments. Effective feedback mechanisms are important for reinforcing the contextual learning and monitoring the students' progresses throughout the learning process.
R5	Respondent 5 indicates the necessity of prioritizing the teacher's role in giving instructions during the teaching and learning process. Especially when teaching the students that are categorized as introverts, who may struggle with social interactions. They are often hesitant to ask and answer the questions in a traditional classroom environment. Hence, providing additional support and attention to these students are essential, as they are able to participate openly during the learning progress. Additionally, the educational game applications that integrate Artificial Intelligence (AI) offer a potential solution by providing the feature of a virtual tutor that represents a teacher in a virtual world. These AI-driven systems enable the introverted students to express their queries and engage with the learning contents without pressuring them for direct classroom interaction. The use of AI technology provides an alternative method to assist the introverted students in overcoming the communication barriers that occur in the traditional classroom environment. Establishing this clarity of the virtual tutor feature embedded in the game development should also align with pedagogical objectives. Hence, it ensures to effectively complement the efforts of real-life educators to be implemented in the form of virtual tutor, rather than replacing the role of teacher itself. Thus, the use of AI technology fosters personalized learning experience in which, aligns with educational practices to enhance the digital and language literacy as well.
R6	Respondent 6 indicates the importance of incorporating the engaging visuals and vibrant colours in the teaching and learning process. It shows that these elements significantly enhance the students' interest and captivate their attention too. The interactive visuals are integrated into the touch screen mobile technologies, such as tablets and smartphones, students are more inclined towards the engagement with the learning content. This strategy of learning can be particularly effective in promoting interactivity with better visuals that align with the learning objectives. Especially, when compared to the conventional learning of using textbooks, which may lead to boredom and lack of engagement among the students. Consistent and structured practice in mastering the language literacy skills by using technology nowadays is crucial to enhance the learning experience. However, R6 raises concerns about the alignment of digital educational game content development with the Malay language curriculum. If the learning content is not properly synchronized with the curriculum, the effectiveness of the educational activities may be significantly compromised. It may lead to inconsistency of engagement during the learning session and hinder the learning objectives to be achieved. Therefore, it is important to highlight the digital learning application to be developed specifically on curriculum alignment. This is to ensure that the learning contents are engaging and pedagogically relevant. Thus, the educators can optimize the use of technology in the classroom as the learning tool that accommodates the diverse learning preferences and needs of the students.

E. Final Specifications of User Requirements

Following a comprehensive analysis of data collection through observation and interview sessions, the final user requirements are systematically refined and simplified. As adaptive digital game-based learning has gained momentum as a key to educational strategy, user requirements are acquired through a comprehensive analysis that reflects the specific needs of users, as shown in Table III.

TABLE III. SPECIFIC USER REQUIREMENTS OF DIGITAL GAME-BASED LEARNING DEVELOPMENT

Category	Requirements	Features
Character design	Human-form character as the player's profile. Human-form character as the virtual tutor representative (NPC).	Customization on player's profile. Allow users to choose and edit the avatar based on their preferences.
Environment design	Various learning environments of jungle, space, zoo and underwater. Immersive learning experience through a well-designed gaming environment.	Provide the users with different background themes.
Gameplay	Provide interactive and experiential aspects of game design to achieve the game goals. Provide game elements to enhance the user experience during the gameplay.	Rules Game mechanics such as game levels, challenges and rewards. Mini games. Navigation menus. Instructional texts and voice Exploration game genre (adventure)
Feedback mechanisms	Provide information to players about their performance, progress tracking and understanding the learning contents. To guide and maintain the engagement of the users.	Explicit feedback Implicit feedback Social feedback Visual feedback Instant feedback Haptic feedback
Use of colours	Influence the player's emotion and user experience. Provide colours and tones on the characters, objects, buttons and environment design.	Bright colours Pastel colours Gradient colours Additive and subtractive colours
Adaptive learning contents	Following the course syllabus of Malay language for primary school students.	Reading Writing Speaking
Functionality	Include animation, pop-up messages. Interactive buttons. Images and icons as non-functional static elements	Display animation Display pop-up messages Buttons Audio

Category	Requirements	Features
	only Include listening and speaking practices.	
Interaction	Touch interaction	Touchscreen technology: Tablet or smartphone
User Experience	Enable to track the performance of adaptive learning.	Progress tracking Leaderboards
Optimization	Integrate Artificial Intelligence (AI) for NPC. Provided automated reply to the player	Virtual tutor (teacher)

V. Discussion

Through the proposed user-centered design method, this research study emphasizes the importance of acquiring the specific user requirements of primary school students by conducting the observation, interview and checklists approaches that are significantly highlighted in the research study [78]. This study employs a mixed-methods approach by combining qualitative and quantitative data to obtain a comprehensive understanding of user preferences for designing and developing the digital game-based learning.

Similarly, this study contributes to the language literacy skills by providing learning tools such as incorporating the adaptive learning features into digital games. Hence, it provides the immersive and effective learning experience in a gamified learning environment for the students to master the language proficiency [47], [67], [68]. Therefore, the integration of personalized learning of the students enables the systems to track students' performances based on varying levels of skills. As a dynamic approach to learning, adaptive learning contents should be aligned with the curriculum to ensure the relevant and structured learning experiences correspond to the academic standards and objectives for students [42], [57], [66], [77]. This study also helps to improve language literacy by integrating adaptive learning features into digital games. The students will be provided with engaging and effective ways of gamified learning approaches to practice and strengthen their language literacy skills.

Therefore, the usability problems of game design led to resulting poor satisfaction among the students to improve the language literacy during the gameplay of digital game-based learning [43], [44], [58], [79]. Thus, there is a need to improve the game design and development based on the students' preferences and needs by acquiring their user specifications. There are 10 categories highlighted components for emphasizing the enhancement of adaptive digital game-based learning for language acquisition, which are character design, environment design, gameplay, feedback mechanisms, use of colours, adaptive learning contents, functionality, interaction, device, user experience, optimization of artificial intelligence. Each requirement specification that is listed has different features to support the causes of usability problems that are related to game design and development [6], [30], [48], [53], [64], [67], [77]. Poorly designed game mechanics can prevent learners from focusing on educational content. Besides, the usability issues in game design can reduce student satisfaction and limit the effectiveness of digital game-based learning for enhancing language literacy skills among them.

As a result, the findings of this research study concludes that students are more inclined to gaming environments rather than focusing on conventional learning due to interactivity where it offers them a more enjoyable interactive learning environment to be experienced. In fact, active interaction between the students and the system developed has beneficial impacts on improving their communication and digital literacy, specifically through the integration of advanced touch interaction technology that provides more features and functionalities for the students to enhance the language learning skills. Therefore, this user specifications highlighted will also be used to develop the conceptual framework for future research. This conceptual framework development will be validated through the expert evaluation and final user testing from both educators and students.

VI. Conclusion

A user-centered design method is essential in understanding and acquiring the preferences and needs of primary school students for language proficiency through digital game-based learning. The involvement of students from the design phase enables them to emphasize the user specifications that are aligned to develop the digital game-based learning based on their learning styles for improving the language literacy skills and cognitive skills too. In fact, the enhancement of learning outcomes should be emphasized on the development of digital game-based learning that provides feedback mechanisms, clear instructions, good navigation of application, uncluttered user interface design, use of colours, environment and character designs, artificial intelligence optimization, gameplay, user interaction and user experience.

Furthermore, the user-centered design approach plays a critical role in identifying usability challenges and enables developers to iteratively refine digital game-based learning tools based on students' feedback. This process ensures that learning experiences are closely aligned with students' preferences, needs, and learning styles, which can enhance both engagement and learning outcomes effectively.

Effective educational games rely not only on UCD but also on the integration of adaptive learning elements. By emphasizing the learning contents, feedback mechanisms, and gamified environment strategies to individual learners, the system can provide targeted guidance, correct misconceptions in real time, and support progression at varying skill levels. Hence, it ultimately contributes to improving language literacy and supporting cognitive development among the students.

For future research, the conceptual framework will be further developed and validated to guide the design of adaptive digital game-based learning for language literacy. Studies could focus on empirically testing its effectiveness across diverse learning style of the students, examining the long-term impact on language proficiency, cognitive development, and engagement to improve the educational strategies.

Acknowledgment

This work was supported by Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia.

REFERENCES

[1] Esteban, A. J., "Theories, Principles, and Game Elements that Support Digital Game-Based Language Learning (DGBLL): A Systematic Review," *International Journal of Learning, Teaching and Educational Research, Society for Research and Knowledge Management*, vol. 23, no. 3, pp. 1–22, 2024.

[2] Isti Septianing, Lina Melati, Nabila Deo Cantika, & Wannuraniza Destiani. ,"Pengaruh Penerapan Game Based Learning Terhadap Motivasi Belajar Siswa Sekolah Dasar," *Khatulistiwa: Jurnal Pendidikan Dan Sosial Humaniora*, vol. 4, no. 4, pp. 4–103, 2024.

[3] Li, Q., Juithong, S., & Sangway, S., "Educational Game Teaching Methods on Communicative English Listening and Speaking Abilities for Early Childhood Students", *International Journal of Sociologies and Anthropologies Science Reviews*, vol. 4, no. 3, pp. 207–218, 2024

[4] Ardila Otero, C. F., Pérez Rueda, L. V., Ballesteros Peña, M. C., & Torres-Barreto, M. L., "Design and implementation of a game-based activity to promote learning of game theory", *Región Científica,* 2023.

[5] Bunt, L., & Bunt, B., "Devise. Tailor. Motivate. A theoretical framework for optimising game-based learning potential and selection", 2023.

[6] Ongoro, C. A., & Fangjiang, Y. Y., "Digital Game-Based Technology for English Language Learning in Preschools and Primary Schools: A Systematic Analysis", *IEEE Transactions on Learning Technologies,* 2023.

[7] Song, C., Shin, S.-Y., & Shin, K.-S., "Optimizing Foreign Language Learning in Virtual Reality: A Comprehensive Theoretical Framework Based on Constructivism and Cognitive Load Theory (VR-CCL)", *Applied Sciences*, vol.13, no. 23, pp. 12557, 2023.

[8] Breien, F., & Wasson, B., "eLuna: A Co-Design Framework for Narrative Digital Game-Based Learning that Support STEAM", *Frontiers in Education*, vol. 6, 2022.

[9] Chew, P. J. ;, & Amir, F. P., "Tahap Penguasaan Kemahiran Literasi Menulis Bahasa Melayu dalam Kalangan Murid Tahun Satu Level of Literacy in Writing Skills in the Malay Language among Year One Pupils", vol. 14, no. 1, pp. 74–87, 2023.

[10] Gillon, G., "Supporting Children Who Are English Language Learners Succeed in Their Early Literacy Development", *Folia Phoniatrica et Logopaedica*, vol 75, no. 4, pp. 219–234, 2023.

[11] Rochanaphapayon, T., "Digital Literacy: Enhancing English Reading Comprehension Among Foreign Language Students", *English Language Teaching*, vol. 17, no. 1, pp. 26, 2023.

[12] Sinar, T. S., Budiman, M. A., Ganie, R., & Rosa, R. N., "Students' Perceptions of Using Roblox in Multimodal Literacy Practices in Teaching and Learning English", *World Journal of English Language*, vol. 13, no. 7, pp. 146–153, 2023.

[13] Babbar, M., & Gupta, T., "Response of educational institutions to COVID-19 pandemic: An inter-country comparison", *Policy Futures in Education*, vol. 20, no. 4, pp. 469–491, 2022.

[14] CIOBANU, R.-C., "M-learning and E-learning Educational Solutions Impact in the COVID-19 Pandemic", *Informatica Economica*, vol. 26, no. 3, pp. 64–73, 2022.

[15] Praveen, C., "Covid-19 compelled RASE Model curriculum transaction for teacher education-A study", *International Journal of English Learning and Teaching Skills*, vol.3, no. 3, pp. 2639–7412, 2021.

[16] Yang, J., Peng, M. Y. P., Wong, S. H., & Chong, W. L., "How E-Learning Environmental Stimuli Influence Determinates of Learning Engagement in the Context of COVID-19?", SOR Model Perspective, *Frontiers in Psychology*, vol. 12, 2021.

[17] Bikri, M. A., & Surat, S., "Meneroka Kaedah Pengajaran dan Pembelajaran Luar Talian Murid Luar Bandar Semasa Pandemik Covid-19", *Malaysian Journal of Social Sciences and Humanities (MJSSH)*, vol. 7, no. 8, pp. 1664, 2022.

[18] Naim, S., & Mokodenseho, S., "Implementation of the Virtual Learning Models during the Covid-19 Pandemic: Students' Perspectives and Its Lessons", *Jurnal Kependidikan: Jurnal Hasil Penelitian Dan Kajian Kepustakaan Di Bidang Pendidikan, Pengajaran Dan Pembelajaran*, vol. 8, no. 3, pp. 617, 2022.

[19] Kimkong, H., & Koemhong, S., "Online learning during COVID-19: Key challenges and suggestions to enhance effectiveness", 2021.

[20] Othman, S., Omar, R., & Sabil, A. M., "Pandemic Covid-19: Impact to Malay Language Learning among the Students in Public Universities", *International Journal of Human Resource Studies*, vol.11, no. 4S, pp. 205, 2021.

[21] Ong, S. L., Peter, M., Zaini, N. A., Ibrahim, K., & Ling, J., Fostering Stem Literacy through Educational Robotics: An Experiential Learning via Stem Carnival, 2024, [Online]. Available: https://www.researchgate.net/publication/378076091.

[22] Gui, Y., Cai, Z., Yang, Y., Kong, L., Fan, X., & Tai, R. H., "Effectiveness of digital educational game and game design in STEM learning: a meta-analytic review", *International Journal of STEM Education*, vol. 10, no. 1, 2023.

[23] Ramli, F. N. A., & Mohamad Arsad, N., "STEM TEACHER DIGITAL LITERACY: RELATIONSHIP BETWEEN DIGITAL LITERACY AND TECHNOLOGY INTEGRATION IN TEACHING AND LEARNING POST COVID-19", *Journal of Nusantara Studies (JONUS)*, vol. 9, no. 2, pp. 316–333, 2023.

[24] Mohamed, R. A. K., & Ali, A. H., "The Effectiveness of using Malay Literary Elements in Primary Education STEM", *International Journal of Academic Research in Progressive Education and Development*, vol. 11, no. 4, 2022.

[25] Mah, D. K., & Groß, N., "Artificial intelligence in higher education: exploring faculty use, self-efficacy, distinct profiles, and professional development needs", *International Journal of Educational Technology in Higher Education*, vol. 2, no. 1, 2024.

[26] Mohammad Aniq Bin Amdan, Naldo Janius, & Mohd Aidil Hazidi Bin Kasdiah., "Concept paper: Efficiency of Artificial Intelligence (AI) tools For STEM Education In Malaysia", *International Journal of Science and Research Archive*, vol. 12, no. 2, pp. 553–559, 2024.

[27] Zhao, Y., Yusof, S. M., Hou, M., & Li, Z., "How Can Generative Artificial Intelligence help Teachers in Early Childhood Education with their Teaching? Analyses from the Perspective of Teaching Methods", *International Journal of Academic Research in Progressive Education and Development*, vol. 13, no. 1, 2024.

[28] Zakari, N. A., Abd. Majid, M. Z., & Hussin, M., "Kecicіran Murid Sekolah di Malaysia: Suatu Pemerhatian Awal", *Malaysian Journal of Social Sciences and Humanities (MJSSH)*, vol. 7, no. 2, 2022.

[29] Banseng, S., Gerry, D., Zarina, P., Khalid, M., & Faaizah, N., "The Influence of Iban Students' Motivation on the Achievement in the Malay Language Subject as a Second Language", *International Journal of Education, Information Technology and Others (IJEIT)*, vol. 6, no. 3, pp. 1–8, 2023.

[30] Shaheen, A., Ali, S., & Fotaris, P., "Assessing the Efficacy of Reflective Game Design: A Design-Based Study in Digital Game-Based Learning", *Education Sciences*, vol. 13, no. 12, 2023.

[31] Bakthaselvan, L., Saiful, M., Hussin, H. Bin, Melayu, A. P., Malaya, U., Wong, K. L., & Yue, S., "PENGINTEGRASIAN TEKNOLOGI MAKLUMAT DAN KOMUNIKASI (TMK) DALAM PENDIDIKAN BAHASA MELAYU: PANDEMIK COVID-19", *In Jurnal Melayu Sedunia*, vol. 5, 2022.

[32] Kadir, A., Md Nor, M., & Hutagalung, F. D., "Penerapan Dan Sumber Halangan Amalan Pendidikan Anti Bias Dalam Pengajaran di Prasekolah Luar Bandar", *Jurnal Penyelidikan Pendidikan*, vol. 39, 2021.

[33] Rosly, N. J., Zainiyah, N., & Mokhtar, Z., "MENDEPANI CABARAN LITERASI PEMBELAJARAN BAHASA MELAYU DALAM KALANGAN MURID ORANG ASLI JAKUN LENGA, JOHOR: SATU TINJAUAN AWAL", In *Asian Journal of Civilizational Studies (AJOCS)*, vol. 3, 2021, [Online]. Available: http://ajocs.com

[34] Parthasarathy, P. K., Mittal, A., & Aggarwal, A., "LITERATURE REVIEW: LEARNING THROUGH GAME-BASED TECHNOLOGY ENHANCES COGNITIVE SKILLS", *International Journal of Professional Business Review*, vol. 8, no. 4, 2023.

[35] Arabiat, D., Al Jabery, M., Robinson, S., Whitehead, L., & Mörelius, E., "Interactive technology use and child development: A systematic review", *Child: Care, Health and Development*, vol. 49, no. 4, pp. 679–715, 2022.

[36] Jeckie, V., & Hanee Ariffin, A., "Interactive Mobile Application in Learning Iban Language", *International Journal of Recent Technology and Applied Science*, vol. 2, pp. 81–89, 2023.

[37] Schmid, R., Pauli, C., & Petko, D., "Examining the use of digital technology in schools with a school-wide approach to personalized learning", *Educational Technology Research and Development*, vol. 7, no. 2, pp. 367–390, 2023.

[38] Kuzmina, Y. V., Avdeeva, S. M., Tarasova, K. V., Popova, A. V., & Bitsiokha, Y. A., "Digital Literacy, Cognitive Control and Student Use of Digital Devices", *Psychological Science and Education*, vol. 28, no. 4, pp. 81–97, 2023.

[39] Reddy, P., Chaudhary, K., & Hussein, S., "A digital literacy model to narrow the digital literacy skills gap", *Heliyon*, vol. 9, no. 4, 2023.

[40] Zheng, Y., Zhang, J., Li, Y., Wu, X., Ding, R., Luo, X., Liu, P., & Huang, J., "Effects of digital game-based learning on students' digital etiquette literacy, learning motivations, and engagement", *Heliyon*, vol. 10, no. 1, 2023.

[41] Shanthi, G., & Renugadevi, A., "DIGITAL LITERACY AMONG PRIMARY SCHOOL TEACHERS", *SHODHSAMHITA*, vol 9, no. 3, 2022.

[42] Chiotaki, D., Poulopoulos, V., & Karpouzis, K., "Adaptive game-based learning in education: a systematic review", In *Frontiers in Computer Science,* Frontiers Media SA, vol. 5, 2023.

[43] Fadhli, M., Kuswandi, D., Utami, P. S., Sartika, S. B., & Hardyman Bin Barawi, M., "Game-Based Learning and Children's Digital Literacy to Support Pervasive Learning: A Systematic Reviews", vol. 25, no. 3, pp. 386–393, 2023.

[44] Sun, L., Kangas, M., & Ruokamo, H., "Game-based features in intelligent game-based learning environments: a systematic literature review", *In Interactive Learning Environments*. Routledge, 2023.

[45] Vnucko, G., & Klimova, B., "Exploring the Potential of Digital Game-Based Vocabulary Learning: A Systematic Review", *In Systems MDPI,* vol. 11, no. 2, 2023.

[46] Fernando, P. A., & Premadasa, H. K. S., "Use of gamification and game-based learning in educating Generation Alpha: A systematic literature review", *Educational Technology & Society*, vol. 27, no. 2, pp. 114–132, 2023.

[47] Udeozor, C., Abegão, F. R., & Glassey, J., "Measuring learning in digital games: Applying a game-based assessment framework", *British Journal of Educational Technology*, 2023.

[48] Ebadi, S., Rasouli, R., & Mohamadi, M., "Exploring EFL learners' perspectives on using Kahoot as a game-based student response system", *Interactive Learning Environments*, vol. 31, no. 4, pp. 2338–2350, 2023.

[49] Reis, M. H., Almeida, A. M., & Lelis, C., "Empowering Creativity and Feedback: Lessons Learned from the Development of an App to Assist Game-Based Learning Activities", pp. 491–510, 2023.

[50] Venkatesan, R. K., Banakou, D., Slater, M., & M, M., "Haptic feedback in a virtual crowd scenario improves the emotional response", *Frontiers in Virtual Reality*, vol. 4, 2023.

[51] Ellis, R., "Explicit and Implicit Oral Corrective Feedback", *In The Cambridge Handbook of Corrective Feedback in Second Language Learning and Teaching*, Cambridge University Press, pp. 341–364, 2021.

[52] Mayo, M. del P. G., & Milla, R., "Corrective Feedback in Second versus Foreign Language Contexts", *In The Cambridge Handbook of Corrective Feedback in Second Language Learning and Teaching,* Cambridge University Press, pp. 473–493, 2021.

[53] Kristensen, J. K., Torkildsen, J. v. K., & Andersson, B., "Repeated mistakes in app-based language learning: Persistence and relation to learning gains", *Computers and Education*, vol. 210, 2024.

[54] Cintamulya, I., Mawartiningsih, L., & Warli, W., "The Effect of Optimizing Digital and Information Literacy in Writing Scientific Articles on Students' Critical Thinking Skills", *AL-ISHLAH: Jurnal Pendidikan*, vol. 15, no. 2, pp. 1987–1998, 2023.

[55] Duarte Correia, J., Henriques, S., Raquel, S., Fernandes, G., Abelha, M., Fernandes, S., & Seabra, F., "Literacy(ies) and skills in times of digital education: Exploring communication and interaction in digitally mediated learning worlds", 2023, [Online]. Available: https://www.researchgate.net/publication/372335789.

[56] Rochanaphapayon, T., "Digital Literacy: Enhancing English Reading Comprehension Among Foreign Language Students", *English Language Teaching*, vol. 17, no. 1, pp. 26, 2023.

[57] Benton, L., Mavrikis, M., Vasalou, A., Joye, N., Sumner, E., Herbert, E., Revesz, A., Symvonis, A., & Raftopoulou, C. (2021). "Designing for "challenge" in a large-scale adaptive literacy game for primary school children", *British Journal of Educational Technology*, vol. 52, no. 5, pp. 1862–1880, 2021.

[58] Chowdhury, M., Dixon, L., Kuo, L.-J., Donaldson, J. P., Eslami, Z., Viruru, R., & Luo, W., "Digital game-based language learning for vocabulary development", *Computers and Education Open*, vol. 6, 2024.

[59] Pérez-González, C., Ortega-Sánchez, D., Ishak, S. A., Hasran, U. A., & Din, R., "Media Education through Digital Games: A Review on Design and Factors Influencing Learning Performance", 2023.

[60] Lai, P., Stroud, F., Paladino, A., & Kalamir, N., "Investigating learning designers' perceptions of student cognitive engagement in online learning", *ASCILITE Publications*, 2023.

[61] Nadeem, M., Oroszlanyova, M., & Farag, W., "Effect of Digital Game-Based Learning on Student Engagement and Motivation", *Computers*, vol. 12, no. 9, 2023.

[62] Zhang, Y., "Improvement of Learning Engagement of Primary School Students by Gamification", In *Journal of Education, Humanities and Social Sciences RETPS*, 2023.

[63] Nguyen, H. H., & Nguyen, V. A., "Personalized Learning in the Online Learning from 2011 to 2021: A Bibliometric Analysis", *International Journal of Information and Education Technology*, vol. 13, no.8, pp. 1261–1272, 2023.

[64] Putra Pratama, M., Sampelolo, R., Lura, H., Toraja, I., Toraja, T., Selatan, S., & Id, I. M. A., "REVOLUTIONIZING EDUCATION: HARNESSING THE POWER OF ARTIFICIAL INTELLIGENCE FOR PERSONALIZED LEARNING", *Klasikal: Journal of Education, Language Teaching and Science*, vol. 5, 2023.

[65] Gaurav, D., Kaushik, Y., Supraja, S., Yadav, M., Gupta, M. P., & Chaturvedi, M., "Empirical Study of Adaptive Serious Games in Enhancing Learning Outcome", *International Journal of Serious Games*, vol. 9, no. 2, pp. 27–42, 2022.

[66] Miraz, M. H., Ali, M., & Excell, P. S., "Cross-cultural usability evaluation of AI-based adaptive user interface for mobile applications", *Acta Scientiarum - Technology*, vol. 44, 2022.

[67] Nizam, D. N. M., Rudiyansah, D. N., Tuah, N. M., Sani, Z. H. A., & Sungkaew, K., "Avatar design types and user engagement in digital educational games during evaluation phase", *International Journal of Electrical and Computer Engineering*, vol. 12, no. 6, pp. 6449–6460, 2022.

[68] Tuah, N. M., Nizam, D. N. M., & Sani, Z. H. A., "Modelling the Player and Avatar Attachment based on Student's Engagement and Attention in Educational Games", *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, pp. 353–360, 2021.

[69] Yang, L., & Li, R.," Contextualized Game-Based Language Learning: Retrospect and Prospect", *Journal of Educational Computing Research*, 2023.

[70] Govender, T., & Arnedo-Moreno, J., "An analysis of game design elements used in digital game-based language learning", *Sustainability (Switzerland)*, vol, 13, no. 12, 2021.

[71] Saleem, A. N., Noori, N. M., & Ozdamli, F., "Gamification Applications in E-learning: A Literature Review", *Technology, Knowledge and Learning*, vol. 27, no. 1, pp. 139–159, 2022.

[72] Mao, P., Cai, Z., Wang, Z., Hao, X., Fan, X., & Sun, X., "The effects of dynamic and static feedback under tasks with different difficulty levels in digital game-based learning", *Internet and Higher Education*, vol. 60, 2024.

[73] Du, X., & Wang, X., "Play by Design: Developing Artificial Intelligence Literacy through Game-based Learning", *Journal of Computer Science Research*, vol. 5, no. 4, pp. 1–12, 2023.

[74] Lipipun, V., & Angasinha, C., "EDUCATIONAL TRANSFORMATION IN THE DIGITAL ERA: A DEVELOPING

COUNTRY PERSPECTIVE", *Corporate and Business Strategy Review*, vol. 4, no. 4 (Special Issue), pp. 252–261, 2023.

[75] Muenz, T. S., Schaal, S., Groß, J., & Paul, J., "How a Digital Educational Game can Promote Learning about Sustainability", *Science Education International*, vol. 3, no. 4, pp. 293–302, 2023.

[76] Wagan, A. A., Khan, A. A., Chen, Y.-L., Yee, P. L., Yang, J., & Laghari, A. A., "Artificial Intelligence-Enabled Game-Based Learning and Quality of Experience: A Novel and Secure Framework (B-AIQoE)", *Sustainability*, vol. 15, no. 6, pp. 5362, 2023.

[77] Røed, R. K., Baugerud, G. A., Hassan, S. Z., Sabet, S. S., Salehi, P., Powell, M. B., Riegler, M. A., Halvorsen, P., & Johnson, M. S., "Enhancing questioning skills through child avatar chatbot training with feedback", *Frontiers in Psychology*, vol. 14, 2023.

[78] Preece, J., Sharp, H., & Rogers, Y., *Interaction Design: Beyond Human-Computer Interaction* (6th ed.). Wiley, 2023.

[79] Okta Riyandi, Mhd. A., Santoso, H. B., & Hadi Putra, P. O., "The Application of Game Mechanics and Technological Trend in Game-Based Learning: A Review of the Research", *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, vol 7, no. 4, pp. 774–781, 2023.

[80] Abidin, S. R. Z., Noor, S. F. M., Ashaari, N. S., & Yadegaridehkordi, E., "Improving literacy skills using the serious game model compatible with brain-based learning", *Multimedia Tools and Applications*, 2024.

[81] G Mingyu, G., Md Yunus, M., & M. Rafiq, K. R., "Educational Games and Game-based Approaches in Higher Education: A Systematic Review (2014-2023)", *International Journal of Academic Research in Progressive Education and Development*, vol 13, no. 1, 2024.

[82] Puaadi, F. A., & Abdul Aziz, A., "The Implementation of Game-Based Learning in Enhancing Learners' Vocabulary in Primary ESL Classroom", *International Journal of Advanced Research in Education and Society*, vol. 6, no. 2, pp. 817–828, 2024.

[83] Zakaria, A., Addanan, F. A., Khalili, N. N. M., Zakaria, Y. K., & Mohamad, M., "A Systematic Literature Review of the Impact of Game-based Learning on English Learners", *International Journal of Academic Research in Business and Social Sciences*, vol. 14, no. 8, 2024.

[84] Mohamad, M., Ibrahim, N. A. E., Nasaruddin, N., & Zakaria, N. Y. K., "The Impact of Gamified Learning Techniques in Improving Writing Skills outcomes: A Systematic Review", *International Journal of Academic Research in Business and Social Sciences*, vol. 14, no. 8, 2024.

Towards More Effective Automatic Question Generation: A Hybrid Approach for Extracting Informative Sentences

Engy Yehia[1], Neama Hassan[2], Sayed AbdelGaber[3]
Department of Information Systems-Faculty of Commerce and Business Administration, Helwan University, Cairo, Egypt[1, 2]
Department of Information Systems-Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt[3]

***Abstract*—Informative Sentence Extraction (ISE) is one of the crucial components in Automatic Question Generation (AQG) and directly influences the quality and relevancy of the generated questions. Instructional texts often contain not only informative but also irrelevant sentences. This results in the creation of poor-quality or distorted questions when irrelevant, non-informative sentences have been used as input. Therefore, the basic problem discussed in this paper is how to provide a systematic method for filtering out such sentences and retaining those that are pedagogically valuable. The purpose of ISE is to filter out irrelevant, low-quality information and retain only the factually dense sentences, express key concepts and are contextually significant. This paper proposes a hybrid approach for extracting informative sentences that combines lexical, statistical, and semantic criteria to identify informative sentences suitable for generating educational questions. The proposed approach consists of two modules: the first module employs four techniques in order to evaluate the informativeness of sentences, which are the keyword-based scoring, Named Entity Recognition (NER), information gain (IG) and Sentence-BERT (SBERT). The second module utilizes multiple fusion strategies to integrate the results derived from the informative sentence extraction techniques. The preprocessed sentences extracted from educational materials were ranked and filtered based on their informativeness coverage. The evaluation results indicate that the hybrid approach can improve the extraction of informative sentences rather than using individual methods. Such a contribution is important for enhancing the performance of downstream tasks in AQG systems, such as distractor generation and question formulation.**

Keywords*—Automatic Question Generation (AQG); informative sentence extraction; NER; SBERT; question answering; information gain; fusion strategies

I. Introduction

Automatic Question Generation (AQG) has attracted considerable interest in the field of educational technology and has provided scalable assessment, intelligent tutoring and personalized learning environment applications [1] [2]. An important part of any functional AQG system is the task of Informative Sentence Extraction (ISE). It entails the process that determines textual parts that are semantically rich, pedagogically valuable, and contextually relevant [3]. These sentences are the main component upon which meaningful and high-quality questions are constructed. Improper or inaccurate selection of sentences may decrease the level of coherence and relevancy of generated questions [4], [5].

The main issue addressed in this paper is that the sentence extraction step remains a relatively unexplored bottleneck in AQG pipelines, despite significant advances in natural language generation and deep learning methods. While recent progress in neural language models has substantially improved the fluency, grammaticality, and contextual relevance of generated questions, much less attention has been given to identifying which sentences within instructional material are most informative and pedagogically valuable as question sources. Most current systems use heuristic rules [6] or deep neural networks in an end-to-end methodology where sentence selection is an implicit or black-box procedure. There are possibilities of including redundant, indefinite, and insignificant sentences that adversely influence the importance and quality of the generated questions [4], [7].

The current paper attempts to mitigate this problem by introducing a hybrid approach of informative sentence retrieval that combines shallow linguistic features with deeper semantics. Our approach combines: lexical relevance scoring based on the coverage of educationally relevant keywords and key phrases, Named Entity Recognition (NER) to identify domain-relevant concepts, proper nouns, and entities at the center of instructional content, Information Gain (IG) computations quantifying the discriminative power of each sentence within the informational landscape of a document, and semantic encoding with sentence-Bert [8] that captures word context and relatedness. We implement our methodology on educational texts within information system academic disciplines. The findings demonstrate that the system outperforms existing heuristic or embedding-only baselines in both the quality of sentence selection and downstream performance in question generation. The proposed approach has been optimized to perform effectively with low-resource educational materials, such as PDF textbooks or lecture notes.

The structure of this paper is organized as follows: Section II summarizes the background and relevant studies on informative sentence extraction. Section III presents a hybrid framework for extracting informative sentences in automatic question generation systems. Section IV discusses the experimental setup, results, and evaluation, while Section V discusses the evaluation results. Section VI concludes the paper and outlines potential directions for future research.

II. Background and Related Work

A. Background

The extraction of the informative sentences is one of the most applicable processes in the context of natural language processing that could apply in the education field in an automatic question generator (AQG), answer selection, and educational content summarization. Traditional summarization strategies focus only on conciseness and identifying key components [9], [10], which are insufficient for educational question generation, where these sentences must be both informative and question-worthy. In AQG, it is essential to select sentences that are both semantically dense and academically rich. Traditional methodologies, including term frequency-inverse document frequency (TF-IDF) and Latent Dirichlet Allocation (LDA), were employed to extract significant textual components [11].

Named entity recognition (NER) has been used to provide a semantic aspect to text, enabling the extraction of sentences that may contain information worthy of questioning [12]. However, traditional approaches are frequently unable to provide sufficient information regarding required contextual knowledge of the learning resources. Recent developments in transformer-based models, such as BERT and Sentence-BERT, significantly enhance semantically sensitive representations, enabling reliable meaning extraction across sentences and facilitating deeper filtering, clustering, and classification [8].

Researchers have proven that phrases defining and explaining causal relationships and structured processes form the basis of high-quality educational questions [2], [13]. Effective sentence extraction has to be consistent with learning objectives, as outlined by Bloom's Taxonomy, which assesses students across many cognitive skill levels, from simple recall to analytical reasoning [14]. Consequently, sentence extraction systems have to satisfy cognitive requirements to select not only syntactically perfect but also semantically valuable sentences. Traditional approaches employed shallow principles, such as sentence location or keyword presence [15], whereas current methods use pretrained transformers to evaluate sentence significance [16]. However, neither of these approaches is sufficient independently, especially for domain-specific contexts where linguistic indicators can greatly vary.

B. Related Works

The extraction of informative sentences is a concern addressed from various perspectives, including general text summarization [17], [18], [19], [20], [21] and specifically opinion summarization, where users need concise yet detailed insights [22], [23]. In information retrieval, the selection of informative sentences is crucial for improving information access efficiency and enhancing the retrieval process, particularly for domain-specific contexts, by applying models that improve relevance and reduce redundancy [24], [25]. Customer requirements analysis is the most recent domain under investigation in informative sentence extraction research.

In [26], researchers employed Transformer networks to identify informative sentences that mirror customer needs in user-generated content. However, it does not clearly address Automatic Question Generation (AQG) in its study or findings.

In the context of automatic question generation systems (AQG), various methodologies for the selection of informative sentences have been proposed. Early methods proposed rule-based and stylized filtering heuristics, such as sentence length, the presence of proper names, and POS tag sequences, to recognized sentences that are worthy of being turned into questions in early AQG systems [27]. Although effective to a certain degree, these techniques are characterized by limited generality and insufficient semantic knowledge.

In this study [28], sentences are considered informative if they contain at least one term derived from specified important concepts for generating multiple-choice test items from electronic documents.

In [29], the author combined a set of criteria, including the number of tokens, the number of clauses, the probabilistic context-free grammar score, and well-defined context. Then manually calculate the score of sentences depending on the occurrence of these criteria; if the score is greater than a threshold, select the sentence as informative.

Another study employed candidate rules to identify phrases with a certain form, such as sentences containing definitions, to create a specific type of question [30]. Additional research used machine learning approaches such as the Support Vector Machine (SVM) classifier based on some features such as sentence length, verb domain, named entity, parts-of-speech, chunk, word position, known-unknown word and acronym [31].

Majumder and Kumar suggested an approach based on parse structure similarity and a rule-based technique to identify informative sentences for multiple-choice questions. The algorithm compared the testing content to a pre-compiled reference set of parse structures that are associated with existing multiple-choice questions (MCQs) and then selected the final informative sentences through rule-based post-processing. This method is considered domain-specific, as it requires a set of existing MCQs within the same domain to create the reference set, which is not easily available [32].

In [33], the author proposed a method to identify informative micro aspects in news texts which can be used to generate relevant questions based on the extracted informative aspects by applying semantic role labelling, named-entity recognition, handcrafted rules and machine learning techniques.

A number of investigations focus on employing sentence structure analysis, dependency parsing, and named entity recognition to discern informative target sentences and concepts for question generation from provided sentences [34], [35], [36], [37].

Several studies employed neural models to discover informative phrases, such as [38], which implemented a two-stage system: neural key-phrase identification followed by sequence-to-sequence question creation.

Previous research on informative sentence extraction has investigated various approaches, each with specific methodological limitations as presented in Table I. Rule-based and heuristic approaches, dependent on basic features like

sentence length, keyword presence, or part-of-speech (POS) patterns, indicate restricted semantic depth, inadequate abstraction, and limited domain relevance [27] [30].

TABLE I. SUMMARY OF RELATED WORK ON INFORMATIVE SENTENCE EXTRACTION

Study	Approach	Technique	Key Limitations
[27], [30]	Rule-based and Heuristic	Sentence length, keyword presence, and POS patterns	Limited semantic depth, poor generalization, narrow coverage
[28], [29]	Concept-based Scoring	Predefined concept lists, token count, clause count, and PCFG scores	Manual curation, brittle to unseen terms
[31]	Feature-based ML	SVM classifier with lexical, and syntactic features (length, POS, position…)	Requires annotated training data, extensive feature engineering, limited context capture
[32]	Parse-Structure Matching	Matching parse trees to MCQ reference set	Needs pre-compiled MCQs, Domain-specific, not widely applicable
[33]–[37]	Hybrid Linguistic and ML	SRL, NER, handcrafted rules, and ML models	Computationally intensive, rule-heavy, Complex pipeline
[38]	Neural Models	Transformer-based key phrase, and sequence-to-sequence generation	Requires large training datasets, less interpretable

Concept-based scoring approaches, such as those employing predefined concept lists, token frequencies, or probabilistic context-free grammar (PCFG) scores, involve considerable manual curation and exhibit instability when faced with unique or domain-specific terminology [28] [29]. Feature-based machine learning methodologies, such as support vector machines (SVMs), which use lexical and syntactic features, require annotated datasets and extensive feature engineering while still capturing only a limited amount of contextual information [31]. Parse-structure matching techniques, which correlate sentence parse trees with multiple-choice question (MCQ) references, reveal significant domain dependency, requiring pre-compiled MCQs and presenting constrained scalability [32].

Hybrid linguistic and machine learning pipelines that incorporate semantic role labelling (SRL), NER, manually generated rules, and classifiers frequently demonstrate high processing costs, a large number of rules, and overly complicated structures [33], [34], [35], [36], [37]. Recently, neural methodologies, such as transformer-based key phrase extraction and sequence-to-sequence creation, have demonstrated potential but depend on extensive training datasets and encounter interpretability challenges [38].

Our work provides a hybrid framework, described as domain-adaptive, that combines keyword scoring, named entity recognition (NER), information gain, and SBERT similarity with alternative fusion strategies to ensure robust sentence selection, even when using low-resource educational material. It bridges the gap between shallow heuristics and complex neural systems while keeping interpretability and practical deployability, which addresses the needs of Automatic Question Generation (AQG) systems in the educational environment.

III. METHODOLOGY

The extraction of informative sentences is a significant NLP technique, with particular application in Automatic Question Generation (AQG), answer selection, and content summarization. Informative sentence extraction (ISE) is an essential part of our comprehensive AQG smart model, which is currently under development. During the development of our AQG smart model, we found that the output of informative sentence extraction significantly impacts the quality of the generated questions; therefore, we conducted this research to provide hybrid approaches aimed at optimizing the quality of informative sentence extraction (ISE). This section outlines the suggested methodology, a multi-step hybrid technique designed for the identification of informative sentences specifically for automatic question generation (AQG). This methodology incorporates various supplemental techniques: keyword-based scoring, named entity recognition (NER), information gain (IG) scoring, and contextual semantic filtering using Sentence-BERT (SBERT). Each technique assists in evaluating sentence informativeness from different linguistic and semantic perspectives. Following that, several fusion scoring methods have been used to combine the informativeness scores derived from the four implemented methods, therefore generating an aggregated score to rank sentence informativeness. The hybrid ISE approach includes two modules: Sentence Informativeness Scoring and Fusion Scoring Strategies, as seen in Fig. 1.

The primary input for the hybrid ISE approach includes a database of simple phrases derived from unstructured instructional resources that were preprocessed by the preprocessing module of our AQG smart model before being used in this research. The subsequent subsections present a detailed description of the hybrid ISE approach.

A. Sentence Informativness Scoring

The Sentence Informativeness Scoring Module is designed to evaluate and filter informative content based on its instructional value. This module takes as its input a learning material database composed of preprocessed simple sentences. The output is a refined Sentence Knowledge Database, consisting exclusively of sentences identified as informative, conceptually substantial, and suitable for AQG downstream educational application.

1) Keyword-based scoring: This approach evaluates sentence informativeness through a keyword-based implication to align the text with the instructional domain. A set of domain-specific terms such as DSS, information systems, software, and hardware has been manually selected to cover critical areas of the learning material. Every sentence is evaluated by phrase-level matching to detect both single-word keywords and multi-word phrases in this list. Sentences containing one or more matching words are designated as informative and assigned a keyword score, which corresponds to the quantity of matched terms. The key concepts of this methodology originate from prior studies on automatic question generation, wherein

manually generated keywords effectively assist the selection of key content [39]. Keyword existence is considered one of the heuristic feature-based sentence selections that have been frequently used in educational question generation systems [40], [41].

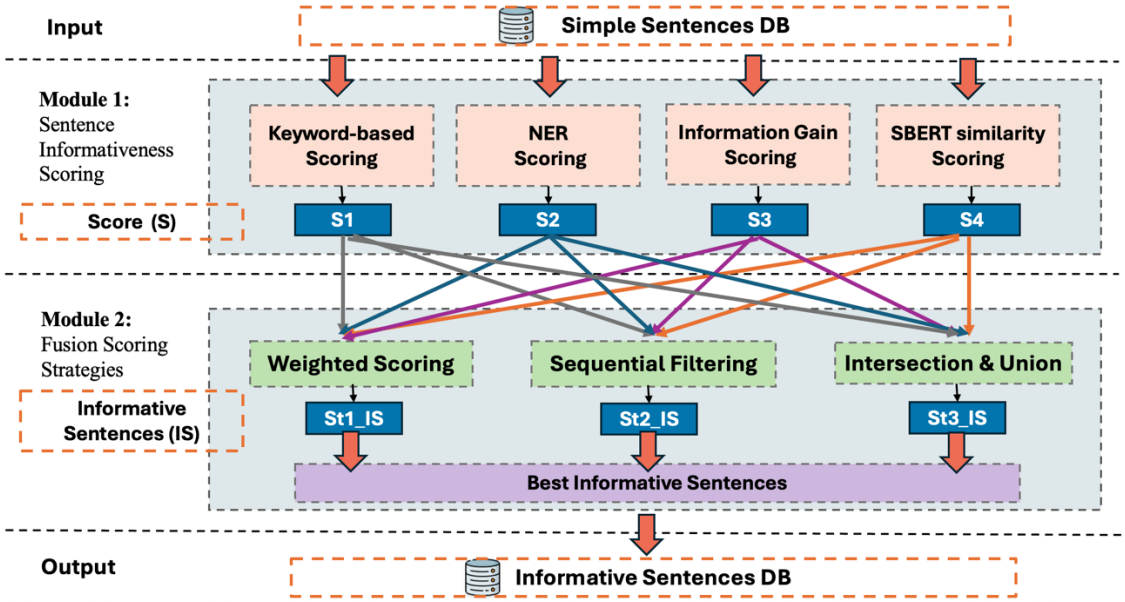


Fig. 1. The proposed hybrid ISE approach.

2) Named Entity Recognition (NER) scoring: Named entity recognition (NER) is a subset of computer science and natural language processing (NLP) that is more directly related to the identification and classification of entities in unstructured text into categories established in advance, such as persons, geographical regions, and organizations [42]. Entity recognition is the process of extraction, disambiguation and connectivity of an entity in a raw text to some useful and structured knowledge bases [43]. Such emphasis allows the generation of more relevant and logical questions adapted to the content [44]. In this approach, the NER technique is used to identify semantically rich sentences using factual and contextual entities. This implies that sentences that refer to individuals, companies, places, or times are most likely to express evaluative information and may be used in the question generation task [45], [46].

SpaCy's lightweight English language model (en_core_web_sm) is used to extract named entities from each sentence in the input corpus. Then the model parses every sentence to identify the entities that exist in it and assign them to the existing types of entities (e.g., PERSON, ORG, DATE, GPE, etc.) [47]. All sentences including at least one named entity are retained. The NER score for each retained sentence is the sum of the number of discovered entities found within it.

3) Information gain scoring: In data and text mining, Information Gain (IG) is a statistical tool that evaluates a term's or word's importance inside a given text. This metric assesses a term's prominence inside a document as well as the likelihood that it falls into a particular category [48]. The provided method is based on the information-theoretic measure of the informativeness of sentences calculated as IG scores. The basic idea is that statements containing statistically significant or relevant terms will serve as more effective question generators, as they are likely to capture fundamental concepts and differences [49]. Text processing starts with tokenizing the simplified sentences corpus into a single word and the identification of the global frequency distribution of all information. In contrast to frequency-based scoring, the IG method utilized here depends on entropy. The reason for the distinction is that words able to reduce the majority of ambiguity, according to the corpus, are considered to represent higher informative quality. Information gain is computed as shown in Eq. (1), (2), and (3).

$$IG(w) = H(C) - H(C|w) \quad (1)$$

$$H(C) = -\sum_{x \in V} P(x)\ log_2 P\ (x) \quad (2)$$

$$IG(s) = \frac{1}{|s|}\sum_{w \in s} IG(w) \quad (3)$$

where, *H(C)* denotes the entropy of the full corpus of sentences C. It is computed as the negative sum over all possible vocabulary labels $x \in V$, where V is the set of vocabulary and $P(x)$ is the probability of observing vocabulary x. The conditional entropy, $H(C|w)$, represents the remaining uncertainty in corpus given the presence/absence of word w. $IG(w)$ is the reduction in entropy achieved by conditioning on that word. A sentence s is considered as a sequence of words $\{w1,w2,...,w|s|\}$, where $|s|$ denotes the number of words in the sentence. The information gain of the sentence, $IG(s)$, is calculated as the average of the information gain values of its constituent words. Algorithm 1 presents the IG scoring steps. This approach was implemented by the tokenisation with the Natural Language Toolkit (NLTK) and an entropy-based

model applied to obtain the IG scores [50]. Sentences are then ranked in a descending order according to their IG scores.

Algorithm 1: Information Gain (IG) Scoring for Sentence Informativeness

Input: A corpus C of sentences

Output: A list of sentence-level information gain scores $IG(s_i)$ for each sentence $si \in C$

Step 1: Tokenize all sentences and compute global word frequencies $f(w_j)$.

Step 2: Calculate corpus entropy $H(C)$.

Step 3: For each word w_j, compute conditional entropy $H(C|w_j)$.

Step 4: Compute word-level information gain $IG(w_j)$.

Step 5: For each sentence s_i, compute sentence-level score $IG(s_i)$ as the average $IG(w_j)$ of its words.

Step 6: Return all sentence scores $\{IG(s_i)\}$.

4) SBERT-based semantic relevance scoring: Sentence-BERT (SBERT) is a variant of BERT, which represents semantically informed sentence embeddings that enable efficient similarity calculations and thus dramatically lower the computation costs [8]. Large pretrained language models, such as BERT and Sentence BERT, provide effective sentence embeddings that are highly correlated with human similarity ratings [51]. SBERT was suggested to learn sentence representation only by performing the computation on a single query question rather than on pairs of sentences. This model is effective for semantic textual similarity (STS) [52]. To improve the informativeness rank, we used Sentence-BERT (SBERT) as a semantic representation method.

SBERT, unlike other shallow linguistic features such as word frequency (count) or keyword matching, SBERT enables us to encode once and for all our sentences into a dense vector that encapsulates the context in which it appears. This enables semantic comparison, beyond simple lexical overlap, which proves particularly useful in educational texts [8]. The algorithm is performed in multiple steps. First, a pre-trained SBERT was used to encode the fixed-length embedding of each sentence of the input corpus. Second, the average of all embeddings was used to compute a semantic centroid vector, which would represent a general semantic space of the instructional material. Third, cosine similarity was used to estimate a scalar informativeness score between each sentence embedding and the centroid as shown in Eq. (4), (5), and (6).

$$\mathrm{sim}(s_i, s_j) = \frac{v_i \cdot v_j}{|v_i||v_j|} \quad (4)$$

$$c_i^* = \frac{1}{n-1} \sum_{j \neq i} \mathrm{sim}(s_i, s_j) \quad (5)$$

$$c_i = \frac{c_i^* - \min(C^*)}{\max(C^*) - \min(C^*)} \quad (6)$$

Where the raw centrality score of a sentence, denoted by c_i^*, is the average semantic similarity of sentence s_i. c_i is the normalized centrality score. The collection of all raw centrality scores across the corpus is represented by $C^*=\{c_1^*, c_2^*, \ldots, c_n^*\}$, with $\min(C^*)$ and $\max(C^*)$ denoting the minimum and maximum values within this set, respectively, for use in the normalization step. The pairwise similarity between two sentences, $sim(s_i, s_j)$, is computed as the cosine similarity between their embeddings. v_i and v_j represent the dense vector embeddings of sentences s_i and s_j generated using a pre-trained SBERT model. The Euclidean norms of these vectors, $|v_i|$ and $|v_j|$, are used to scale the cosine similarity, while the dot product $v_i \cdot v_j$ constitutes its numerator. Algorithm 2 presents the SBERT-Based Semantic Centrality Scoring steps.

Algorithm 2: SBERT-Based Semantic Centrality Scoring

Input: A set of candidate sentences $S=\{s_1, s_2, \ldots, s_n\}$.

Output: A vector of normalized semantic centrality scores $C=\{c_1, c_2, \ldots, c_n\}$.

Step 1: Encode each sentence $s_i \in S$ into a dense vector representation v_i using a pre-trained Sentence-BERT (SBERT) model.

Step 2: Compute the pairwise semantic similarity $sim(s_i, s_j)$ between all sentence embeddings.

Step 3: For each sentence s_i, calculate its raw centrality score c_i^*, as the average similarity with all other sentences in S.

Step 4: Normalize all raw centrality scores to the range $[0,1]$.

Step 5: Return the normalized semantic centrality scores $C=\{c_1, c_2, \ldots, c_n\}$.

Sentences having higher scores are expected to have more central and representative information, while sentences with lower scores might be less informative. The result of this step is a ranked list of sentences, and each has a numerical value corresponding to the degree of semantic proximity to the distribution of the overall content. This ranked output is further utilized in the downstream level of the fusion scoring strategies to choose informative sentences.

B. Fusion Scoring Strategies for Informative Sentence

To exploit the complementary properties of the individual scoring techniques (NER, keyword-based scoring, IG, and SBERT), we suggested and evaluated three distinct fusion strategies. Each strategy calculates the final informativeness score by integrating the outputs of the primary methods in different ways, thereby providing varied trade-offs among precision, recall, and semantic coverage.

1) Weighted scoring: In this strategy, each of the four methods is used as one complementary signal within a unified scoring model. Each score is normalised in the range [0, 1]. Both NER and keyword scores are taken as binary, while IG and SBERT scores are taken as continuous. The final score is calculated as:

$$\mathrm{FinalScore}(s) = w_{\mathrm{NER}} \cdot \mathrm{NER}(s) + w_{\mathrm{KW}} \cdot \mathrm{Keywords}(s) + w_{\mathrm{IG}} \cdot \mathrm{IG}(s) + w_{\mathrm{SBERT}} \cdot \mathrm{SBERT}(s) \quad (7)$$

where, $w_{\mathrm{NER}} + w_{\mathrm{KW}} + w_{\mathrm{IG}} + w_{\mathrm{SBERT}} = 1$

This strategy depends on the distribution of weight to make a balance between precision and recall. For example, a higher weight on SBERT prioritizes semantic richness, while increasing the weight on NER ensures factual bases.

2) Sequential filtering: The sequential filtering approach sequentially filters the results of different selection processes; at each stage, it eliminates a portion of possible sentences. Such an approach reduces informative sentences to the most

informative and pedagogically salient ones. As presented in Fig. 2, sequential filtering strategy works as follows:

a) Stage 1:Boolean filtering: This initial stage applies a logical filter to retain only sentences with high potential for informativeness:

- Filtering with NER: The process first discards those sentences that do not have named entities. Sentences with numerous entities (e.g. persons, places, organizations, or whatever is relevant to the domain) are most likely to carry information that can be evaluated as factual and conceptually significant [53]. Entity sentences are also quite useful in providing domain-specific and factual questions [1].
- Filtering with Keyword: In parallel, domain-specific keywords are used to retain sentences that directly reference core course concepts. This step ensures that essential and pedagogically valuable content is preserved early in the process.

b) Stage 2: Ranking_based filtering: The reduced candidate subset from stage1 is then used in:

- IG, SBERT ranking: The reduced candidate subset from stage1 is then ranked using a combined score derived from Information Gain (IG) and SBERT semantic similarity scores after normalization. IG highlights sentences that introduce statistically distinctive information within the corpus [54], while SBERT measures their semantic density and relevance. The fusion of these metrics prioritizes sentences that are relatively novel, semantically rich, and aligned with the instructional objectives.
- Last Selection: Top informative sentences are selected from previously ranked scoring as the final informative set.
- In this strategy, Sentences that match many criteria are given priority at each of the filtering stages. This multilevel approach has made the selection process precise by ensuring that the chosen sentences contain many entities and are informative regarding context and domain-specific knowledge. Sequential filtering is also used to reduce the noise and improve downstream AQG tasks.

3) Set-based combination (intersection/union): This strategy addresses sentence selection as a task of set-based combinations instead of a cross-methods approach. This technique outlines two mechanisms: Intersection and Union as shown in Fig. 3.

- Intersection (I): All those sentences that have failed to correspond to at least 2 methods or more are removed. The accuracy of this practice has been enhanced because only sentences that have been agreed upon by multiple methods are retained where precision has been observed from many perspectives. Intersection-based models have been employed in multi-criteria decision-making and have delivered positive outcomes in terms of inspiring false positives in text mining processes [55]. The purple central overlap in Fig. 3 highlights the intersection region.

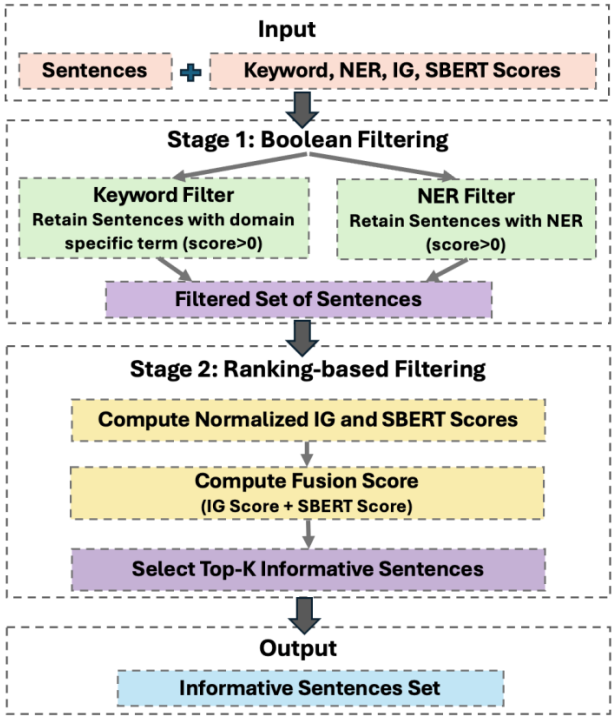


Fig. 2. Sequential filtering strategy steps.

- Union (U): The sentences identified by at least one method are considered and kept in the final selection. This strategy gives the greatest emphasis to recall (maximization of coverage), which is desired when answering a large number of different types of questions, as one may want to cover a large set of sentence candidates in an education setting [2]. The dashed rectangle in Fig. 3 outlines the Union region.

Typically, the set-based composition also involves a type of flexibility to apply an intersection- or union-based approach that is ranged between precision and recall.

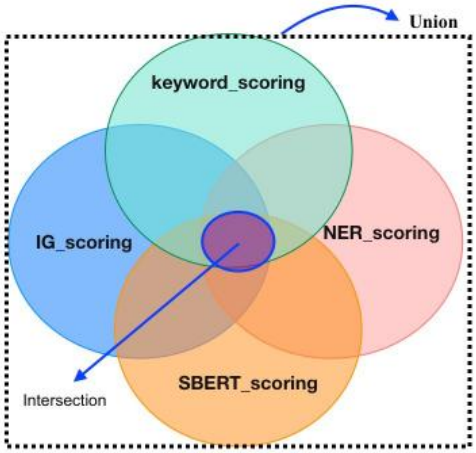


Fig. 3. Intersection and union strategies.

IV. Experimental Setup and Results

This section outlines the experimental setup, results, and evaluation applied to measure the effectiveness of the proposed hybrid ISE approach.

A. Tools and Dataset

The experimental implementation was conducted in Python 3.10 using open-source NLP libraries. Text preprocessing and NER were performed with NLTK and spaCy, while word-level IG scores were computed through a custom entropy-based implementation to estimate sentence informativeness. For semantic modelling, SBERT embeddings were obtained using the Sentence-Transformers library with the all-MiniLM-L6-v2 model, which provides efficient contextual sentence-level representations. All development was carried out in the PyCharm IDE [56], and experiments were executed on a macOS Monterey machine with an Apple M1 8-core processor and 16 GB RAM, eliminating the need for external high-performance computing resources.

The experimental dataset used to test the hybrid approach was generated from instructional materials for an information systems course taught in the business information systems program at Helwan University in Egypt. The dataset sentences are extracted from a book chapter covering core concepts such as hardware, software, and processes. After applying the preprocessing pipeline, which is part of our smart AQG model mentioned before, the dataset was transformed into a sentence-level corpus consisting of approximately 460 sentences. A subset of 58 candidate sentences was selected for evaluation, and a gold-standard reference set was manually annotated by two domain experts, labelling each sentence either informative or non-informative. A sample of the sentences dataset is represented in the tables that located in the following sections.

B. Sentence Informativness Scoring Results

In this module, each sentence in the preprocessed educational dataset was evaluated using the four informative sentence scoring techniques outlined in the proposed approach, to capture different dimensions of informativeness. The keyword-based scoring method assigned a score to each sentence based on the number of matched keywords it contained. Table II provides an example result of the keyword-based scoring method. Sentences such as S1, received higher scores (score = 3) because they contain several domain-specific words (e.g. software, hardware, and information system). Likewise, statements describing system components such as S3 and S5 received moderate scores (score = 2), whereas those using only a single keyword (e.g. “agile”) attained a score of 1. In contrast, phrases without any keywords, such as S40 received a score of 0, indicating poor informativeness according to this criterion.

The second technique applied in the suggested approach is NER, which detects factual entities in text and acts as a strong indicator of informative content. Table III displays a sample of the output produced by this method. As illustrated, sentences involving multiple entities earned higher scores, such as in S58, which earned a score of 3 due to the identification of entities. Sentences containing two entities, such as S8, received a score of 2, whereas those with a single defined entity, such as S19, were assigned a value of 1. On the other hand, sentences without any of the identifiable items, such as S42, received a score of 0.

TABLE II. Sample of the Keyword Based Scoring Output

Index	Sentence	Matched Keywords	Keyword Score
S1	Information systems are combinations of hardware, software, and telecommunications networks.	software, hardware, information system	3
S3	Computers, keyboards, disk drives, iPads, and flash drives are all examples of information systems hardware.	hardware, information system	2
S5	Software is a set of instructions that tells the hardware what to do.	software, hardware	2
S36	Agile methods emphasize flexibility in IS development.	agile	1
S40	This chapter served as an introduction to key IS themes.	-------	0

TABLE III. Sample of the Output Produced by NER-Based Scoring Method

Index	Sentence	Entities with Labels	NER Score
S58	In 1995, Jeff Bezos expanded Amazon from an online bookstore into a global e-commerce and cloud computing leader.	1995 (DATE), Jeff Bezos (PERSON), Amazon (ORG)	3
S8	Examples of operating systems include Microsoft Windows on a personal computer and Google’s Android on a mobile phone.	Microsoft Windows (ORG), Google (ORG)	2
S19	We will discuss processes in Chapter 8.	Chapter 8 (LAW)	1
S42	Big data technologies enable real-time analysis.	--------	0

The third technique applied is IG scoring, which evaluates the informativeness of sentences through examining the contribution of individual words inside every sentence, as illustrated in Eq. (1), (2), and (3). Table IV displays sample outcomes of IG-based scoring. S14 received the highest score (0.327), indicating that it has words with significant discriminatory power within the dataset. Sentences such as S41 and S47 received comparatively high scores; however, S47 did not clarify the basic concepts of the instructional material. In contrast, lines such as S40 received lower scores (0.235), indicating a reduced concentration on high-IG words.

The final technique applied in the proposed approach is SBERT-based semantic centrality scoring, which evaluates informativeness at the semantic level by using the pre-trained all-MiniLM-L6-v2 SBERT model. Semantic centrality has been evaluated, and the centrality score for each sentence was calculated as illustrated in Eq. (4), (5), and (6). Sentences with higher centrality scores are considered more informative, as they are semantically important to the main topics of the dataset.

Table V displays an example of the outcomes of the SBERT-based testing. The top-ranked sentences are S2 and S1, with each sentence scoring 0.332. Both statements clearly provide fundamental definitions and applications of

information systems, highlighting their significance within the corpus.

TABLE IV. SAMPLE OF THE OUTPUT PRODUCED BY IG-BASED SCORING METHOD

Index	Sentence	IG score
S14	Data is stored and processed by the Information System.	0.327321791
S41	Readers should now recognize the importance of IS in daily and organizational life.	0.271537004
S47	The next chapter will discuss system development in detail.	0.263299162
S5	Software is a set of instructions that tells the hardware what to do.	0.256487202
S40	This chapter served as an introduction to key IS themes.	0.235468327

TABLE V. SAMPLE OF THE OUTPUT PRODUCED BY SBERT-BASED SCORING METHOD

Index	Sentence	SBERT Score
S2	Information systems are essential to modern organizations, supporting daily operations, decision-making, and long-term strategies.	0.332139224
S1	Information systems are combinations of hardware, software, and telecommunications networks.	0.332084447
S16	People include all the individuals who interact with the Information System, from users who input data to IT professionals who maintain and manage the system.	0.293755561
S43	User-centered design ensures systems meet user needs.	0.286940813
S5	Software is a set of instructions that tells the hardware what to do.	0.252976418

Sentences such as S16 received relatively high scores 0.294, indicating their contribution to describing the vital roles and practical relevance of Information Systems. In contrast, sentences such as S5 achieved slightly lower scores 0.253, indicating limited relevance to the overall semantic context of the dataset.

C. Fusion Scoring Strategies Results

This section illustrates the results that were achieved through the fusion scoring strategies module.

1) ST1_weighted scoring: The input for this strategy contains four different scores resulting from the applied methods in module 1. The results have been normalized between 0 and 1 to facilitate integrating them into a final score. We assigned a weighted contribution of (0.4, 0.1, 0.2, 0.3) to each score, respectively, based on the nature of the questions that needed to be extracted from the course material. The highest weight has been assigned to keyword-based scoring, which indicates that in educational text, explicit domain terminology is the most reliable marker of informativeness. NER has the lowest weight, indicating that entity presence alone is not a strong enough criterion in the course domain, such as dates and names. The combined weights must equal one, as presented in Eq. (7). Table VI illustrates how ST1 provides the distinct differences between well-informed and poorly informed information.

TABLE VI. SAMPLE OF THE OUTPUT PRODUCED BY ST1

Index	Sentence	Combined Score
S1	Information systems are combinations of hardware, software, and telecommunications networks.	0.863
S23	Management Information Systems (MIS): This system is used to collect data from different sources, process the information, and present it to the management team for decision making.	0.811
S7	Most models describe information systems as having five main components: hardware, software, data, people, and processes.	0.783
S14	Data is stored and processed by the Information System.	0.659
S3	Computers, keyboards, disk drives, iPads, and flash drives are all examples of information systems hardware.	0.623
S17	Without people, even the most advanced system cannot generate meaningful results.	0.159
S15	For example, your street address, the city you live in, and your phone number are all pieces of data.	0.158
S32	Later sections will revisit ethics in greater detail.	0.107
S35	Sustainability is a new focus, considering environmental impacts.	0.099
S24	Readers should not worry if some terms seem unfamiliar, as they will be explained later.	0.054

The first five sentences (e.g. S1, S23, S7, S14, S2) involve definitions, functional roles, and system structures, which are essential to knowledge acquisition and therefore effective in formulating questions. In contrast, the last five sentences (i.e. S34, S35, 0S24, S27, S30) notably involve historical notes, general insights, and commentary that decrease their informative significance within the context. This indicates that this method effectively prioritizes statements based on their relevance to questions by emphasizing conceptually and semantically rich content.

2) ST2_sequential filtering: In sequential filtering strategy (ST2), the four different scores resulting from the applied methods in module one were used in sequential order.

Sentences were initially reviewed using keyword-based scoring and NER and then fine-tuned using IG and SBERT similarity scores. The filter applied at the first stage was Boolean filtering applied based on the existence of a keyword or NER in sentences to ensure that relevant sentences are kept. This step of filtering reduced the size of the data from 58 to 41 possible sentences, thereby discarding less informative data at an early stage. The second stage used the candidate list from stage 1 to apply a ranking process based on their combined IG and SBERT semantic similarity scores. The combination has enabled the system to test statistical relevance and semantic density, creating a more balanced measure of informativeness. The ranking process also further refined the results in order to end up with the 30 high-quality sentences. Table VII presents samples of these sentences.

3) Set-based combination (intersection/union): The union strategy was firstly applied to keyword and NER methods, which produced a total of 41 candidate sentences. Subsequently, 20 additional sentences were integrated into the existing 41 candidate sentences, utilizing the highest IG+SBERT scores, resulting in a total of 44 unique sentences.

(41+20 → 44 final) indicating that only 4 of the 20 semantically selected sentences didn't selected with the rule-based set and other 16 common was removed to avoid redundancy.

TABLE VII. SAMPLE OF THE OUTPUT PRODUCED BY ST2

Index	Filtered Sentence
S14	Data is stored and processed by the Information System.
S1	Information systems are combinations of hardware, software, and telecommunications networks.
S2	Information systems are essential to modern organizations, supporting daily operations, decision-making, and long-term strategies.
S51	Today, most IS are computer-based, but people and processes remain critical components.
S47	The next chapter will discuss system development in detail.
S3	Computers, keyboards, disk drives, iPads, and flash drives are all examples of information systems hardware.
S23	Management Information Systems (MIS): This system is used to collect data from different sources, process the information, and present it to the management team for decision making.
S16	People include all the individuals who interact with the Information System, from users who input data to IT professionals who maintain and manage the system.
S18	Processes refers to the procedures and protocols that guide the use of the system, including security measures, backup procedures, and data management policies.

This evidence indicates that rule-based filters neglected to recognize certain semantically selected statements, consequently underscoring the complementary nature of semantic scoring. An example would be sentences such as [S54], "During the input stage, data are collected from different sources, such as sensors, transactions, or user input," which would not usually be captured using keyword or Named Entity Recognition but are obtained through semantic scoring. The union strategy generally produces a wider and more informative final set of sentences, resulting from a balanced combination of explicit lexical structure and semantic similarity.

The Intersection strategy produced only 18 sentences, as it required the previous identification of "candidate sentences" by both keyword and NER techniques. The resulting sentences were then ranked using a combination of IG and SBERT scores to balance statistical and semantic relevance in the final ranking. Fig. 4 shows the intersection of the 18 sentences across scoring methods.

When attempting to retrieve the top 20 sentences common to the four applied methods, only 18 sentences were provided due to the constrained size of the intersection set, with a maximum possible of 18. This indicates that intersection procedures prioritise precision, albeit at the expense of recall, since multiple sentences selected by one individual method are excluded.

D. Evaluation Results

The results comparing the performance of the various strategies used to extract informative sentences demonstrate that there are disparities in performance in relation to three measures, namely, precision, recall, and overall F1-score as shown in Eq. (8), (9), and (10).

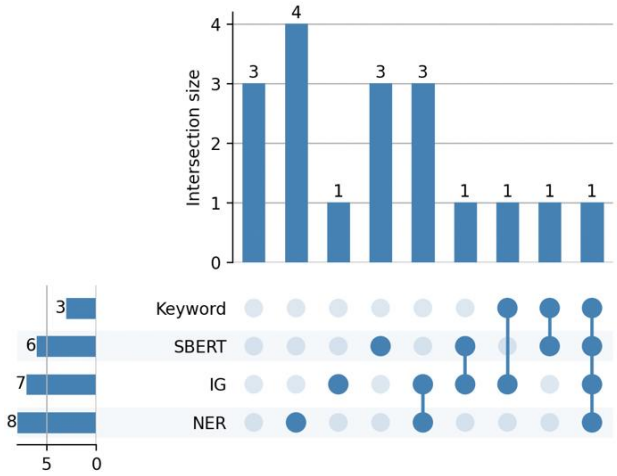


Fig. 4. Intersection of sentences across scoring methods.

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

According to Dalianis (2018), these metrics are defined as follows: Precision is the proportion of correctly identified informative sentences among all those classified as informative. Recall is the proportion of informative sentences successfully retrieved by the system, and the F_1-score is their harmonic mean [57]. In this context:

- True Positives (TP): Candidate sentences correctly classified as informative.
- False Positives (FP): Non-informative sentences incorrectly selected by the system.
- False Negatives (FN): Informative sentences that were missed by the system.

This metric choice balances correctness (precision) and coverage (recall): selecting too few sentences risks missing important question-worthy content, while selecting too many introduces noise into the generated questions.

The findings suggest that the combination of methods usually scores better than single methods. Table VIII and Fig. 5 present the evaluation results of the four individual methods and the fusion scoring strategies. The Sequential method was the most precise (0.967), which can be interpreted as its capacity to identify sentences with the highest correctness, despite the relatively low precision (0.527). In parallel to that, the Weighted Score solution had one of the highest precisions (0.925) yet a more moderate recall (0.673) when compared to the rest of the solutions, leading it to a moderately high F1-score (0.779).

The Union strategy showed the highest overall trade-off, with both a high recall (0.745) and high precision (0.932) and, therefore, the highest F1-score (0.828). Single-component approaches, like NER (F1 = 0.500) and Information Gain (F1 = 0.627) were not effective, too either due to weak recall or

average-level precision. The worst in overall performance was the Intersection strategy (F1 = 0.411), whose restrictive filtering resulted in only a very few numbers of selected sentences and poor recall (0.273). The number of sentences selected also influenced performance, with broader selections, as in the Union method (44 sentences) and Keyword method (40 sentences), support high recall but restrict recall in other methods, such as Intersection (18 sentences) or NER (19 sentences) due to limited coverage. Overall, the results indicate that although high-precision strategies will guarantee the correctness, strategies as Union and Weighted Score that balance between precision and recall are still better to use for enhancing quality and diversity of questions in automatic question generation systems.

TABLE VIII. THE EVALUATION RESULTS OF THE FOUR INDIVIDUAL METHODS AND THE FUSION SCORING STRATEGIES

Strategy	Precision	Recall	F1_Score	Selected Sentences
Keyword	0.700	0.757	0.727	40
NER	0.737	0.378	0.500	19
IG	0.700	0.568	0.627	30
SBERT	0.767	0.622	0.687	30
Weighted_Score	0.925	0.673	0.779	40
Sequential	0.967	0.527	0.682	30
Union	0.932	0.745	0.828	44
Intersection	0.833	0.273	0.411	18

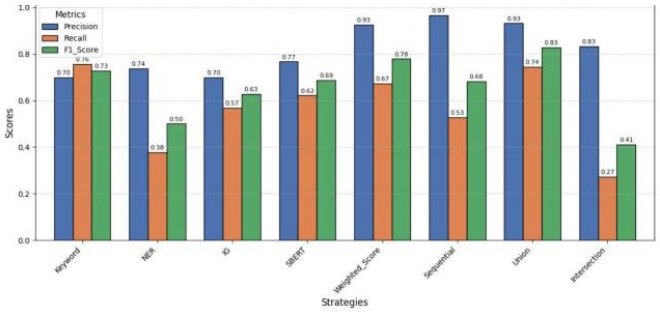


Fig. 5. Comparison of evaluation results.

V. DISCUSSION

This research proposed a multi-step hybrid framework that combines complementary methodologies for evaluating informativeness from linguistic and semantic perspectives. Keyword-based scoring identifies domain-specific lexical factors, named entity recognition (NER) identifies factual and entity-filled content, information gain (IG) scoring determines word-level contributions to sentence informativeness using entropy-based metrics, and Sentence-BERT (SBERT) provides contextual semantic representations to capture deeper meanings.

To overcome the limitations of individual methodologies, different fusion methods have been used to combine scores from these four methods, providing aggregated measures of informativeness for sentence ranking. The findings highlight a significant research shortcoming in existing AQG pipelines: despite major effort to enhance question generation models, the sentence extraction process is inadequately examined and often simplistic, depending only on individual features like keywords or entity presence. The evaluation results provide significant insights into the performance of different methods for extracting informative sentences in automatic question generation.

A significant finding is that individual methodologies, such as Named Entity Recognition (NER) and Information Gain (IG), have constrained efficiency. NER achieved the lowest F1-score (0.500), largely due to its poor recall (0.378), which indicates that while NER can capture entity-rich sentences with relatively satisfactory precision, it misses a substantial proportion of informative content. Likewise, IG (F1 = 0.627) shows decreased recall, indicating that scoring entirely on frequency and entropy fails to adequately cover semantically diverse informative sentences. These limitations highlight that relying solely on isolated linguistic or statistical indicators is insufficient for identifying a broad and reliable set of candidate sentences. The SBERT-based scoring method achieved a reasonably high F1-score of 0.687, surpassing both NER and IG by utilising contextual embeddings that incorporate sentence-level semantics. However, the absence of complementary structural or domain-specific signals continues to limit its effectiveness. In contrast, the examination of fusion-based methods indicates that hybrid approaches can significantly reduce these limitations. Strategies like Weighted Score and Union integrate lexical, statistical, and semantic attributes into a comprehensive metric of informativeness. The Weighted Score technique achieved an F1-score of 0.779 by balancing high accuracy (0.925) with satisfactory recall (0.673), whereas the Union strategy obtained superior overall performance (F1 = 0.828), utilising both strong precision (0.932) and recall (0.745). The results indicate that hybrid evaluation methods not only highlight highly accurate sentences but also offer extensive coverage of potentially informative content, which is essential for supporting diversity in AQG.

Another significant result is that hybrid approaches provide a practical solution to the trade-off between precision and recall, a continuous challenge in information extraction tasks. Whereas highly precise methods such as the Sequential approach minimise noise but at the cost of excluding valuable content, more balanced strategies like Union and Weighted Score ensure that extracted sentences are both accurate and academically rich. This finding highlights the necessity of overcoming rigid, single-dimensional extraction criteria in preference for adaptive frameworks that can incorporate several dimensions of informativeness.

Overall, these outcomes reinforce the key point of this work: the extraction of informative sentences should be treated as a core research problem in AQG, rather than as a secondary or optional task. The demonstrated improvements achieved by hybrid strategies highlight their potential to bridge the gap between technical advances in natural language generation and the practical needs of educational applications, where both the quality and diversity of generated questions are essential.

VI. CONCLUSION

The paper presented a novel hybrid model of Informative Sentence Extraction (ISE) in Automatic Question Generation (AQG) with keyword scoring, Named Entity Recognition, Information Gain, and Sentence-BERT embedding. The integration of complementary methods with fusion techniques substantially enhanced the process of selecting contextually precise and rich informative sentences. Experimental evidence showed that both the Union and Weighted Score strategies would be effective when a balance in precision and recall is required. The results demonstrate the success of multi-method combination in the improvement of question quality and diversity, which will serve as a strong basis for downstream AQG tasks. Generally, the presented framework contributes to the evolution of intelligent learning tools because it optimizes sentence selection based on the unstructured instructional material. Future work should extend these hybrid models by incorporating large language models (LLMs), adaptive weighting mechanisms, and domain-specific knowledge graphs to further improve generalizability and pedagogical validity. Additionally, evaluating the generated questions with learners and educators would provide practical evidence of the impact of hybrid extraction strategies on learning outcomes, thus bridging technical innovation with educational practice.

REFERENCES

[1] Y. Chali and S. A. Hasan, "Towards Topic-to-Question Generation," Computational Linguistics, vol. 41, no. 1, pp. 1–20, Mar. 2015, doi: 10.1162/COLI_a_00206.

[2] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A Systematic Review of Automatic Question Generation for Educational Purposes," Int J Artif Intell Educ, vol. 30, no. 1, pp. 121–204, Mar. 2020, doi: 10.1007/s40593-019-00186-y.

[3] G. Chen, J. Yang, and D. Gasevic, "A comparative study on question-worthy sentence selection strategies for educational question generation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, 2019, pp. 59–70. doi: 10.1007/978-3-030-23204-7_6.

[4] X. Du, J. Shao, and C. Cardie, "Learning to ask: Neural question generation for reading comprehension," in ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), Association for Computational Linguistics (ACL), 2017, pp. 1342–1352. doi: 10.18653/v1/P17-1123.

[5] E. M. Perkoff, A. Bhattacharyya, J. Cai, and J. Cao, "Comparing Neural Question Generation Architectures for Reading Comprehension," in Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 556–566. doi: 10.18653/v1/2023.bea-1.47.

[6] D. Lindberg, F. Popowich, J. Nesbit, and P. Winne, "Generating Natural Language Questions to Support Learning On-Line," in Proceedings of the 14th European Workshop on Natural Language Generation (ENLG), Sofia, Bulgaria: Association for Computational Linguistics, 2013, pp. 105–114. doi: 10.3115/v1/W13-2114.

[7] S. Guo, Y. Guan, R. Li, X. Li, and H. Tan, "Incorporating Syntax and Frame Semantics in Neural Network for Machine Reading Comprehension," in Proceedings of the 28th International Conference on Computational Linguistics, Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 2635–2641. doi: 10.18653/v1/2020.coling-main.237.

[8] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, 2019. doi: 10.18653/v1/d19-1410.

[9] H. P. Luhn, "The Automatic Creation of Literature Abstracts," IBM J Res Dev, vol. 2, no. April, pp. 159–165, 1958.

[10] P. Watanangura, S. Vanichrudee, O. Minteer, T. Sringamdee, N. Thanngam, and T. Siriborvornratanakul, "A Comparative Survey of Text Summarization Techniques," SN Comput Sci, vol. 5, no. 1, 2024, doi: 10.1007/s42979-023-02343-6.

[11] W. A. N. G. Zhuohao, W. A. N. G. Dong, and L. I. Qing, "Keyword Extraction from Scientific Research Projects Based on SRP-TF-IDF," Chinese Journal of Electronics, vol. 30, no. 4, 2021, doi: 10.1049/cje.2021.05.007.

[12] V. Harrison and M. Walker, "Neural generation of diverse questions using answer focus, contextual and linguistic features," in INLG 2018 - 11th International Natural Language Generation Conference, Proceedings of the Conference, 2018. doi: 10.18653/v1/w18-6536.

[13] Y. Lu and S.-E. Lu, "A survey of Approaches to Automatic Question Generation: from 2019 to Early 2021," in Proceedings of the 33rd Conference on Computational Linguistics and Speech Processing (ROCLING 2021), The Association for Computational Linguistics and Chinese Language Processing, Oct. 2021, p. 151.

[14] R. Heer, "A Model of Learning Objectives based on a Taxonomy for Learning, Teaching, and Assessing: A revision of Bloom's Taxonomy of Educational Objectives," Jul. 2012. Accessed: Sep. 15, 2025. [Online]. Available: https://www.celt.iastate.edu/wp-content/uploads/2015/09/RevisedBloomsHandout-1.pdf

[15] M. Heilman and N. A. Smith, "Good question! Statistical ranking for question generation," in NAACL HLT 2010 - Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference, 2010.

[16] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, 2019. doi: 10.18653/v1/d19-1410.

[17] A. Nenkova and K. McKeown, "A Survey of Text Summarization Techniques," in Mining Text Data, Boston, MA: Springer US, 2012, pp. 43–76. doi: 10.1007/978-1-4614-3223-4_3.

[18] Y. J. Kumar, O. S. Goh, H. Basiron, N. H. Choon, and P. C. Suppiah, "A review on automatic text summarization approaches," 2016. doi: 10.3844/jcssp.2016.178.190.

[19] H. N. Fejer and N. Omar, "Automatic multi-document Arabic text summarization using clustering and keyphrase extraction," Journal of Artificial Intelligence, vol. 8, no. 1, 2015, doi: 10.3923/jai.2015.1.9.

[20] Z. Edress and Y. Ortakci, "Optimizing Text Summarization with Sentence Clustering and Natural Language Processing," International Journal of Advanced Computer Science and Applications, vol. 15, no. 10, 2024, doi: 10.14569/IJACSA.2024.01510115.

[21] A. P. Widyassari et al., "Review of automatic text summarization techniques & methods," 2022. doi: 10.1016/j.jksuci.2020.05.006.

[22] L. Zhu, S. Gao, S. J. Pan, H. Li, D. Deng, and C. Shahabi, "Graph-based informative-sentence selection for opinion summarization," in 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013), 2013, pp. 408–412. doi: 10.1109/ASONAM.2013.6785738.

[23] S. and P. S. J. and L. H. and D. D. and S. C. Zhu Linhong and Gao, "The Pareto Principle Is Everywhere: Finding Informative Sentences for Opinion Summarization Through Leader Detection," in Recommendation and Search in Social Networks, A. U. and A. E. Ulusoy Özgür and Tansel, Ed., Cham: Springer International Publishing, 2015, pp. 165–187. doi: 10.1007/978-3-319-14379-8_9.

[24] J.-L. Koh and C.-W. Cho, "Informative Sentence Retrieval for Domain Specific Terminologies," 2011, pp. 242–252. doi: 10.1007/978-3-642-21822-4_25.

[25] K. P. Snehalkumar Nanubhai Patel, "Informative Term Selection And Novel Sentence Extraction," Journal of Engineering Computers & Applied Sciences, vol. 4, no. 6, pp. 165–167, Jun. 2015.

[26] M. Kashi, S. Lahmiri, and O. A. Mohamed, "Comprehensive analysis of Transformer networks in identifying informative sentences containing customer needs," Expert Syst Appl, vol. 273, p. 126785, May 2025, doi: 10.1016/J.ESWA.2025.126785.

[27] M. Heilman and N. A. Smith, "Human Good Question! Statistical Ranking for Question Generation," in Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, Los Angeles, California: Association for Computational Linguistics, Jun. 2010, pp. 609–617. Accessed: Sep. 16, 2025. [Online]. Available: https://aclanthology.org/N10-1086/

[28] R. Mitkov, L. A. Ha, and N. Karamanis, "A computer-aided environment for generating multiple-choice test items," Nat Lang Eng, vol. 12, no. 2, 2006, doi: 10.1017/S1351324906004177.

[29] J. Pino, M. J. Heilman, and M. Eskenazi, "A selection strategy to improve cloze question quality," Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains, 2008.

[30] A. Kurtasov, "A system for generating cloze test items from Russian-language text," in International Conference Recent Advances in Natural Language Processing, RANLP, 2013.

[31] R. Correia, J. Baptista, M. Eskenazi, and N. Mamede, "Automatic generation of Cloze question stems," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012. doi: 10.1007/978-3-642-28885-2_19.

[32] M. Majumder Sujan Kumar Saha, M. Majumder, and S. Kumar Saha, "Knowledge Management & E-Learning Automatic selection of informative sentences: The sentences that can generate multiple choice questions Automatic selection of informative sentences: The sentences that can generate multiple choice questions," Knowledge Management & E-Learning, vol. 6, no. 4, pp. 377–391, 2014.

[33] A. Y. Bokan and T. A. S. Pardo, "Automatic Aspect Identification: The Case of Informative Microaspects in News Texts," Research in Computing Science, vol. 90, no. 1, 2015, doi: 10.13053/rcs-90-1-18.

[34] O. Keklik, T. Tuglular, and S. Tekir, "Rule-based automatic question generation using semantic role labeling," IEICE Trans Inf Syst, vol. E102D, no. 7, 2019, doi: 10.1587/transinf.2018EDP7199.

[35] W. T. Sewunetie and L. Kovacs, "Automatic Question Generation based on Sentence Structure Analysis," in Proceedings of the 2023 24th International Carpathian Control Conference, ICCC 2023, 2023. doi: 10.1109/ICCC57093.2023.10178946.

[36] M. Blšták and V. Rozinajová, "Automatic question generation based on sentence structure analysis using machine learning approach," Nat Lang Eng, vol. 28, no. 4, 2022, doi: 10.1017/S1351324921000139.

[37] M. Blšták and V. Rozinajová, "Automatic question generation based on analysis of sentence structure," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9924 LNCS, 2016. doi: 10.1007/978-3-319-45510-5_26.

[38] S. Subramanian, T. Wang, X. Yuan, S. Zhang, A. Trischler, and Y. Bengio, "Neural Models for Key Phrase Detection and Question Generation," 2017, doi: 10.48550/arXiv.1706.04560.

[39] C. A. Nwafor and I. E. Onyenwe, "An Automated Multiple-Choice Question Generation using Natural Language Processing Techniques," International Journal on Natural Language Computing, vol. 10, no. 02, pp. 1–10, Apr. 2021, doi: 10.5121/ijnlc.2021.10201.

[40] I. Pilán, E. Volodina, and L. Borin, "Candidate sentence selection for language learning exercises: from a comprehensive framework to an empirical evaluation," ArXiv, vol. abs/1706.03530, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:896837

[41] X. Ma, Q. Zhu, Y. Zhou, and X. Li, "Improving Question Generation with Sentence-Level Semantic Matching and Answer Position Inferring," Dec. 2020. doi: 10.48550/arXiv.1912.00879.

[42] I. Keraghel, S. Morbieu, and M. Nadif, "Recent Advances in Named Entity Recognition: A Comprehensive Survey and Comparative Study," arXiv preprint, Dec. 2024, doi: 10.48550/arXiv.2401.10825.

[43] M. Munnangi, "A Brief History of Named Entity Recognition," Nov. 2024. doi: 10.48550/arXiv.2411.05057.

[44] C. Lopez et al., "Recursive Named Entity Recognition," in Studies in Computational Intelligence, 2022. doi: 10.1007/978-3-030-90287-2_2.

[45] H. Kim, J. Yoo, S. Yoon, and J. Kang, "Automatic Creation of Named Entity Recognition Datasets by Querying Phrase Representations," in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 7148–7163. doi: 10.18653/v1/2023.acl-long.394.

[46] I. E. Fattoh, "Semantic Based Automatic Question Generation using Artificial Immune System," Issn, vol. 5, no. 8, 2014.

[47] Explosion AI, "spaCy 101: Everything you need to know," Explosion AI, Berlin, Germany. Accessed: Sep. 15, 2025. [Online]. Available: https://spacy.io/usage/spacy-101

[48] R. ÇEKİK and M. KAYA, "A New Feature Selection Metric Based on Rough Sets and Information Gain in Text Classification," Gazi University Journal of Science Part A: Engineering and Innovation, vol. 10, no. 4, pp. 472–486, Dec. 2023, doi: 10.54287/gujsa.1379024.

[49] A. Mazyad, F. Teytaud, and C. Fonlupt, "Information Gain Based Term Weighting Method for Multi-label Text Classification Task," 2018. [Online]. Available: https://hal.science/hal-01859697v1

[50] S. , K. E. , & L. E. Bird, "NLTK: The Natural Language Toolkit – nltk.probability.FreqDist," Association for Computational Linguistics. Accessed: Sep. 15, 2025. [Online]. Available: https://www.nltk.org/

[51] J. Opitz and A. Frank, "SBERT studies Meaning Representations: Decomposing Sentence Embeddings into Explainable Semantic Features," Oct. 2022, [Online]. Available: http://arxiv.org/abs/2206.07023

[52] T. T. Ha, V. N. Nguyen, K. H. Nguyen, K. A. Nguyen, and Q. K. Than, "Utilizing SBERT For Finding Similar Questions in Community Question Answering," in Proceedings - International Conference on Knowledge and Systems Engineering, KSE, 2021. doi: 10.1109/KSE53942.2021.9648830.

[53] J. Li, A. Sun, J. Han, and C. Li, "A Survey on Deep Learning for Named Entity Recognition," Mar. 2020, doi: 10.1109/TKDE.2020.2981314.

[54] T. M. Cover and J. A. Thomas, Elements of Information Theory. 2005. doi: 10.1002/047174882X.

[55] G. Kou, P. Yang, Y. Peng, F. Xiao, Y. Chen, and F. E. Alsaadi, "Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods," Applied Soft Computing Journal, vol. 86, Jan. 2020, doi: 10.1016/j.asoc.2019.105836.

[56] JetBrains, "PyCharm IDE (Version 2023.3) [Computer software]," JetBrains s.r.o. (or JetBrains s.r.o., Prague, Czech Republic). Accessed: Sep. 15, 2025. [Online]. Available: https://www.jetbrains.com/pycharm/, 2023.

[57] H. Dalianis, "Evaluation Metrics and Evaluation," in Clinical Text Mining, Springer International Publishing, 2018, pp. 45–53. doi: 10.1007/978-3-319-78503-5_6.

An Integrated Evaluation Using Enhanced Panel Factor Model and Machine Learning: Assessing the Level and Structure of Regional Coordinated Development in the Guangdong-Hong Kong-Macao Greater Bay Area

Li Shi[1], Ting Nie[2]*
Guangzhou Huashang College, Guangzhou 511300, China[1, 2]
Guangdong Huashang Institute for Fintech Research, Guangzhou 511300, China[1, 2]

***Abstract*—Regional sustainable and coordinated development has become a central issue in the backdrop of a reshaped global economic landscape. Therefore, it is particularly important to evaluate the level of regional coordinated development effectively. This study aimed to validate and assess the effectiveness of machine learning algorithms and the Enhanced Panel Factor Model for evaluating regional coordinated development. To this end, based on panel data from 11 cities in the Guangdong–Hong Kong–Macao Greater Bay Area for 2005–2023, we constructed a four-dimensional composite indicator system covering economic growth, structural optimization, innovation-driven development, and social development. First, we employ a factor model to achieve dimensionality reduction and extract latent factors. SPSS and the JiekeLi platform are used for visualization, and finally, we combine LASSO regression with linear regression to build predictive models to verify the explanatory power of key factors for regional coordination. The findings indicate that the traditional factor model performs robustly in structural identification, whereas machine learning methods have advantages in variable selection and fitting accuracy. The empirical results show that the overall level of coordination in the Greater Bay Area has steadily improved; however, substantial disparities among cities remain. This study demonstrates a new pathway that integrates econometrics and machine learning for the comprehensive evaluation of regional development levels. It also conducts a comparative analysis of the applicability and effectiveness of these two methods, thereby offering significant theoretical and practical value.**

Keywords*—Comprehensive evaluation; machine learning; regional coordinated development; comparative analysis

I. Introduction

Under the new landscape of the world economy, economic systems exhibit the characteristics of complex systems, evolving from discrete blocks to system linkages and dynamic equilibrium. Building a vibrant and internationally competitive first-class bay area and world-class city cluster, serving as a demonstration zone for deep cooperation between Mainland China and Hong Kong and Macao, and establishing a model for high-quality development is the key mission of the Guangdong–Hong Kong–Macao Greater Bay Area (the Greater Bay Area). The Outline Development Plan for the Guangdong–Hong Kong–Macao Greater Bay Area explicitly calls for building an open, regionally coordinated development community characterized by connectivity and institutional alignment. Establishing a high-quality coordinated development community in the Greater Bay Area carries dual strategic significance against the backdrop of China's transition to high-quality economic development. Regional coordinated development is a dynamic process characterized by the gradual narrowing of internal disparities and achievement of orderly growth. It involves coordinated and adaptive development of economic, social, ecological, and institutional systems that evolve harmoniously. The result of synergistic development is not just a narrowing of the economic output gap, but also a dynamic and balanced development process in which the overall regional economy converges in terms of intrinsic structure and development capacity [1]. A clear assessment of the grid relationships and structures of regional development, along with identifying the grid coordination capacity and stability, is a fundamental basis for exploring pathways to sustainable, coordinated regional development.

Based on the theory of coordinated development, numerous scholars have conducted extensive empirical research on regional coordinated development. These studies mainly focus on measuring the level of regional coordination [1–9] and analyzing the constituent elements of regional coordination [10–22].

Methods for evaluating regional coordinated development levels primarily include the composite system synergy model, the coupling coordination degree model, system dynamics models, and the entropy-weight method.Shi et al. (2025) employed the Dagum Gini coefficient and a composite system synergy model to measure China's nationwide level of new-quality productive forces, and compared levels across the eastern, central, western, and northeastern regions [2]. Li et al. (2023) constructed a system dynamics model to analyze the driving factors of regional growth and the coupling–synergy effects in Yunnan's border areas [3]. Xiong et al. (2022) drawing on the Haken model and provincial panel data, measured the synergy level of the composite system linking

fintech and the green transformation of manufacturing in China [4]. Huang et al. (2020) adopted a composite synergy model to examine the orderliness and composite system synergy degree between the regional innovation input system and the innovation output system in Henan Province [5]. Ma (2019) used output synergy indicators and applied a composite system synergy model to measure the level of economic coordinated development in the Beijing–Tianjin–Hebei region [6]. Chen et al. (2022) utilized the entropy-weight method and exploratory spatial data analysis (ESDA) to assess the high-quality development levels of seven urban agglomerations along the Yangtze River Economic Belt from 2009 to 2018, and employed geographically weighted regression to analyze influencing factors [7]. Yi et al. (2022) investigated the sustainable development performance of cities in Liaoning Province following a multi-criteria decision-making process, proposing a novel synergetics-based weighting method that accounts for interrelationships among sustainability criteria [8]. Zhang et al. (2021) used a coordination degree index to measure the degree of coordinated development across China's four major metropolitan areas from 2008 to 2019 [9].

In reviewing methods for assessing regional coordinated development levels, the composite system synergy model is the most widely used approach. Meanwhile, a variety of methods are continually being explored depending on the research objectives. The application of machine learning has yet to be substantively reflected in these literatures.

Analyses of regional coordination elements encompassed industrial synergy, innovation-factor synergy, efficiency synergy, comprehensive evaluation of regional high-quality development levels, and assessments of regional coordinated development levels. Ding and Zhao (2025) measured the level of industrial synergy across 16 metropolitan areas in China for the period 2013–2023 [10]. Zhou et al. (2025) examined the economic effects of different industrial coordination models within regions, including co-constructed industrial parks, headquarters economy, and industrial-chain integration [11]. Sun et al. (2025) measured the level of related industrial diversity among the Beijing–Tianjin, Beijing–Hebei, and Tianjin–Hebei subregions [12]. Qin et al. (2025) analyzed the level of regional coordinated development between the innovation chain and the industrial chain during the integration process of the Beijing–Tianjin–Hebei region, as well as the underlying drivers [13]. Lu et al. (2024) measured the level of new-quality productive forces and regional disparities across 30 provincial-level regions in China [14]. Yan et al. (2022) constructed an indicator system spanning three domains—economy and society, environment, and resource carrying capacity—to analyze the level of high-quality development and regional disparities in the Yellow River Basin [15]. Xue and Cai (2022) employed a multilevel factor analysis approach to evaluate China's overall and regional levels of high-quality development [16]. Li and Ci (2025), using panel data for 30 Chinese provinces, investigated the coupling and coordination relationships among the digital economy, carbon emission efficiency, and high-quality economic development [17]. Chang (2023) applied the entropy-weight method to measure the level of coordinated development in the Beijing–Tianjin–Hebei region across five dimensions: economy, society, ecology, government governance, and coordinated development [18].

From the above literature, analyses of regional coordination elements and structures either concentrate on the synergistic relationships among key elements—such as industry, innovation, and the digital economy—or focus on evaluating the composite level of regional aggregate indicators. Systematic assessments of regional system structures are relatively scarce.

Since the issuance of the Outline Development Plan for the Guangdong–Hong Kong–Macao Greater Bay Area, scholars have explored the coordinated development of the GBA from perspectives such as coordination mechanisms and the effectiveness of industrial synergy. Ye et al. (2022) analyzed the evolution of the GBA's coordinated development stages and the shifts in coordination mechanisms under the contexts of globalization, regional governance systems, and technological innovation [19]. Gao (2022) assessed the degree of industrial synergy in the GBA from the viewpoints of industrial planning and spatial layout [20]. Li (2024) examined inter-urban functional differences and structures in the GBA through the lens of collaborative technological innovation [21]. Wang (2025) constructed a dual-factor linkage network using enterprise investment and liner shipping data to analyze the strength of intercity linkages and synergistic effects within the Bay Area [22]. However, a systematic and quantitative evaluation of coordinated development in the GBA has not yet been clearly demonstrated.

Drawing upon the synthesized literature, current methods for measuring regional coordinated development remain primarily grounded in traditional econometric models, with machine learning algorithms not yet widely applied. Research on synergistic elements within regional systems tends to emphasize validation of single (pairwise) relationships among elements, without considering the systemic interactions among multiple intra-regional factors. Therefore, this study employs an improved spatiotemporal global factor analysis model to quantify the level of multi-element coordinated development within regions, identify the structure of the coordination network, and, concurrently, verify the applicability and effectiveness of machine learning algorithms in research on regional coordinated development.

II. Theoretical Analysis and Research Hypotheses

A. Theoretical Analysis

1) Theory of regional coordinated development: The theory of regional coordinated development holds that various elements such as the economy, population, industry, institutions, and infrastructure should be effectively aligned and interconnected among cities or units within a region, thereby promoting the optimal allocation of resources and improving overall efficiency. Regional coordination is not simply the sum of individual city developments; rather, it emphasizes functional complementarity, structural optimization, and the sharing of development outcomes among cities. At its core, it is a dynamic evolutionary process characterized by multidimensional and integrated

optimization. In the case of the Great Bay Area, the region features a unique combination of "one country, two systems," three distinct customs territories, and diverse governance frameworks. These characteristics make coordinated development both challenging and strategically significant. Therefore, it is essential to systematically identify and analyze the evolutionary process and structural level of coordination using scientifically grounded assessment methods.

2) Complex systems theory and the logic of multi-factor matching: From the perspective of systems science, regional development is an open complex system composed of multiple subsystems, such as economic, innovation, social, and ecological systems, which evolve through a pathway of "coupling–coordination–optimization." Systems coordination theory emphasizes that when the interactions among these subsystems become harmonized, the overall order and efficiency of the system are enhanced. In the context of the Great Bay Area, such system coordination is reflected in aspects such as industrial chain collaboration among cities, sharing of innovation resources, and mechanisms for talent mobility. Analyzing a single dimension (such as industry or innovation) in isolation overlooks the complex coupling logic among subsystems. Therefore, adopting a method capable of integrating multiple dimensions and capturing the overall coordination dynamics is of significant importance.

B. Research Hypotheses

To gain an in-depth understanding of the evolutionary trend and structural dynamics of regional coordinated development in the Great Bay Area, this study, grounded in the theories of regional coordinated development, system coordination, and the spatio-temporal global factor analysis model, proposes the following research hypotheses:

- Hypothesis H1: The level of regional coordinated development in the Great Bay Area shows an upward trend over time.
- Hypothesis H2: The level of regional coordinated development in the Great Bay Area can be explained by several latent factors with inherent structural characteristics, forming a structural framework for coordinated development.
- Hypothesis H3: Machine learning methods are applicable to measuring and evaluating the level of regional coordination.

III. Research Design

A. Construction of the Evaluation Index System

The study of synergistic development usually includes the study of synergistic development model (process) and the study of the level (result) of synergistic development. The level of economic synergistic development refers to the result of it. This paper mainly evaluates the level of synergistic development (result) of cities in the Greater Bay Area from four perspectives: the degree of economic growth, the degree of structural optimization, the degree of innovation drive, and the degree of social development to construct the evaluation indicator system of the level of synergistic development of Guangdong, Hong Kong and Macao Greater Bay Area (as shown in Table I).

TABLE I. Evaluation Indicator System of the Level of Synergistic Development of Guangdong, Hong Kong and Macao Greater Bay Area

Primary Indicators	Secondary Indicators	Unit	Symbol	Direction
Degree of Economic Growth	GDP Sum	RMB 100 million	X_1	+
	Fiscal Revenue	RMB 100 million	X_2	+
	Fiscal Expenditure	RMB 100 million	X_3	+
	Total Investment in Fixed Assets	RMB 1 million	X_4	+
	Total Import & Export	100 million USD	X_5	+
Degree of Structural Optimization	Value Added by Secondary Industry as A Percentage of GDP	%	X_6	+
	Value Added by Tertiary Industry as A Percentage of GDP	%	X_7	+
Degree of Innovation Drive	Expenditure on Scientific Research	RMB 10,000	X_8	+
	RMB 10,000	PCS	X_9	+
	Actual Use of Foreign Capital	10,000 USD	X_{10}	+
Degree of Social Development	Per Capita Disposable Income of Urban Residents	RMB	X_{11}	+
	Total Number of Students in Higher Education	Per	X_{12}	+
	Education Expenditure	RMB 10,000	X_{13}	+
	Number of Sickbeds Per 10,000 People	Sheet	X_{14}	+
	Number of Practicing Physicians Per 10,000 People	Per	X_{15}	+

B. Data Sources and Processing

This study employs data from 15 evaluation indicators across 11 cities in the Great Bay Area over the period 2005–2023. Data sources included the Guangdong Statistical Yearbook, statistical yearbooks of the respective cities, the website of the Census and Statistics Department of the Hong Kong SAR Government, the website of the Statistics and Census Service of Macao, and the Wind database. Missing data were supplemented using the incremental mean and median imputation methods.

C. Model Construction

1) Enhanced panel factor analysis model: Factor analysis is a multivariate dimensionality-reduction technique that extracts common factors to explain the principal variation patterns of the original variables. The enhanced panel factor analysis model extends the traditional factor analysis approach. Because the traditional factor analysis model cannot reduce the dimensionality and misses more information on the explanatory variables, Lin (2006) [23] proposed an improved factor analysis model based on the traditional factor analysis model, which replaces "the covariance matrix of the error random vector as a diagonal matrix" in the traditional model with the maximum variance contribution of the factor explanatory variables, thus solving the above-mentioned problems. In 2009, Lin Haiming further studied the application steps of factor analysis model L and provided comprehensive evaluation steps for initial factor analysis and rotated factor analysis, respectively [24]. Zhao (2019) and Shi (2020) provided application cases and improved comprehensive evaluation steps when factor analysis model L was applied to cross-sectional data and time series data [25]-[26]. Therefore, this study adopts an Enhanced Pannel Factor Model to evaluate the level of coordinated economic development in the Great Bay Area. The factor analysis model is given by Eq. (1) as follows:

$$X_{it} = \Lambda F_t + \epsilon_{it} \quad (i = 1, \dots, N;\ t = 1, \dots, T) \qquad (1)$$

where, Xit represents the vector composed of all observed variables at spatial location i and time point t, Λ is the factor loading matrix, F denotes the global common factor vector, and ϵit is the spatiotemporal-specific residual vector.

The specific evaluation steps are as follows:

- Data Preprocessing: Global positive direction transformation and standardization of evaluation indicators. All indicators in this sample are positive indicators. For positive indicators, apply min-max normalization (see Eq.(2)):

$$x_{ijt}^{*} = \frac{x_{ijt} - \min(x_{ijt})}{\max(x_{ijt}) - \min(x_{ijt})} \qquad (2)$$

where, x_{ijt}^{*} represents the data of the i city for the j-th indicator in year t. The normalized data are then further standardized using standard deviation normalization as follows, see Eq. (3):

$$z_{ij} = \frac{x'_{ij} - \mu_j}{\sigma_j} \qquad (3)$$

Where μ_j is the mean of the j-th indicator, and σ_j is the standard deviation. Both are calculated over all regions and all time periods.

- Determination of applicability of factor analysis model: The Pearson correlation coefficient matrix R is calculated for all variables to assess whether a factor structure exists, as shown in Eq. (4). Let r_{jk} denote the correlation coefficient between the j-th and k-th indicators, then:

$$r_{jk} = \frac{\sum_{i=1}^{n}\sum_{t=1}^{T}(x_{ijt} - \bar{x}_j)(x_{ikt} - \bar{x}_k)}{\sqrt{\sum_{i=1}^{n}\sum_{t=1}^{T}(x_{ijt} - \bar{x}_j)^2} \cdot \sqrt{\sum_{i=1}^{n}\sum_{t=1}^{T}(x_{ikt} - \bar{x}_k)^2}} \qquad (4)$$

If there is a simple linear correlation coefficient between indicators exceeding 0.8, it indicates that there is a common factor between indicators, and a factor analysis model is applied. Furthermore, the Kaiser-Meyer-Olkin (KMO) test and Bartlett's test of sphericity are employed to evaluate whether the shared variance among variables is statistically significant, thereby confirming the appropriateness of factor analysis.

- The principal factors are extracted using the maximum variance method (which maximizes the explanatory power of the factors). Suppose m factors are to be extracted, such that the cumulative explained variance reaches a predefined threshold (e.g. $\geqslant$ 85%). After extraction, an orthogonal rotation method such as Varimax is applied to enhance the interpretability of the factors, resulting in a rotated factor loading matrix Λ^{*}.

- Estimation of Factor Scores and Comprehensive Score Calculation: The regression method is used to estimate the score of each sample on each factor. Let F_{ikt} denote the score of city i in year t on factor k. The comprehensive score is then calculated as a weighted sum of the factor scores, see Eq. (5).

$$S_{it} = \sum_{k=1}^{m} w_k F_{ikt} \qquad (5)$$

$$w_k = \frac{\lambda_k}{\sum_{j=1}^{m} \lambda_j} \qquad (6)$$

where, the weights being the variance contribution rate of each factor λ_k, see Eq. (6).

- Calculate the scores of the factors, the scores of the combined factors and the ranking: Calculate and rank the score F_i for t×n sample factors and the composite factor score Fsum ==$\alpha_1 F_1 + \alpha_2 F_2 + \dots + \alpha_k F_k$ (α_i is the variance contribution of each factor).

2) Application of machine learning: A Comparative Analysis of Linear Regression and LASSO Regression Models.

To further test the validity and explanatory power of the common factors extracted through factor analysis and to investigate the explanatory ability of factor scores on the level of regional coordinated development, this study introduces two

machine learning methods: linear regression and Least Absolute Shrinkage and Selection Operator Regression (LASSO) regression to validate and compare the modeling results of factor analysis. This approach not only helps to uncover the linear relationship between factor scores and the level of coordinated development but also provides data support for variable selection, thereby enhancing the interpretability and robustness of the model. The specific applications of this method are as follows:

- Linear regression Modeling: The principal factor scores extracted through factor analysis were used as explanatory variables, while the level of regional coordinated development served as the dependent variable. A multiple linear regression model was constructed to verify the statistical significance and explanatory validity of the common factors identified through the factor analysis.
- LASSO Regression: Variable Selection and Model Simplification: Given the potential multicollinearity among certain factor scores, ordinary multiple linear regression may perform unreliably in high-dimensional settings. To enhance explanatory efficiency and achieve feature compression, this study introduces the Least Absolute Shrinkage and Selection Operator (LASSO) regression model for comparison. By incorporating an L1-norm penalty into the loss function, LASSO shrinks some regression coefficients to zero, thus accomplishing variable selection and parameter estimation simultaneously. This preserves model interpretability while effectively reducing complexity and mitigating overfitting.

D. Software Platforms and Tools

This paper uses several advanced software platforms and tools for analysis and modeling, selected for their robustness, user-friendliness, and compatibility with the research objectives.

1) Jiekeli intelligent modeling platform: The Jiekeli Intelligent Modeling Platform integrates various data preprocessing, statistical modeling, and machine learning algorithms. With its high level of automation, interactive visualization, and powerful computing capabilities, the platform provides strong support for complex data modeling and interpretation of the results in this study. Specifically, the main functions applied in this research include automated data preprocessing, which automatically handles missing value imputation, standardization, normalization, and other basic preprocessing tasks, thereby improving data quality and modeling efficiency. Support for linear modeling and variable selection: Equipped with built-in modules for multiple linear regression and LASSO regression, the platform facilitates efficient modeling between factor scores and regional coordinated development levels, helping identify key variables and enhance model interpretability. Result visualization and model evaluation: It offers a variety of visual tools, such as goodness-of-fit plots, residual distribution charts, and coefficient path diagrams, making it easier to evaluate model performance and interpret results, thus increasing the transparency and rigor of the research.

2) SPSS Analytical tool: As a standard tool for professional statistical analysis, SPSS offers powerful capabilities for data reduction and latent structure exploration through its factor-analysis module. With an intuitive graphical interface, SPSS supports exploratory factor analysis (EFA), which was mainly used for the following purposes: variable reduction: core common factors were extracted using Principal Component Analysis (PCA) or Maximum Likelihood Estimation (MLE) to effectively reduce variable dimensionality; structural validity testing: the suitability of the data was assessed through correlation coefficients (r), the Kaiser-Meyer-Olkin (KMO) measure, and Bartlett's test of sphericity; factor rotation optimization: the software provides various rotation methods, such as Varimax and Direct Oblimin, to enhance the interpretability of the extracted factors.

TABLE II. VARIANCE CONTRIBUTION EXPLAINED BY EACH ROTATED FACTOR

Factor	Variance Contribution λ_i	Variance Contribution (%)	Cumulative Variance Contribution (%)
1	5.047	33.646	33.646
2	3.298	21.988	55.633
3	3.053	20.350	75.984
4	1.494	9.961	85.945
5	1.436	9.576	95.521

IV. EMPIRICAL RESULTS AND ANALYSIS

A. Enhanced Panel Factor Structure Analysis

This study adopts the comprehensive evaluation steps of the enhanced panel factor analysis model to assess the synergistic development level of the Great Bay Area. The dataset comprises p = 15 indicators, denoted as X_1 to X_{15}, covering t = 19 years from 2005 to 2023, with n = 11 cities sampled each year. All indicators X_1 to X_{15} are positive, so no direction adjustment is required. These indicators are globally standardized, with the standardized variables denoted as x_1 to x_{15}. The simple linear correlation coefficients among the indicators were calculated; in particular, the correlations between X_1 and X_2, X_3, and X_4 are 0.874, 0.914, and 0.916, respectively, all exceeding 0.8, indicating the presence of common factors suitable for factor analysis. The rotated factor loading matrices $\boldsymbol{L}_3^{\Gamma},\ldots,\boldsymbol{L}_{15}^{\Gamma}$ were calculated, from which $\boldsymbol{L}_3^{0},\boldsymbol{L}_3^{\Gamma},\ldots,\boldsymbol{L}_{15}^{\Gamma}$ were obtained, and the arithmetic means of the maximum absolute values of each row's elements are denoted as $l_3^{0},l_3^{\Gamma},\ldots,l_{15}^{\Gamma}$.

The explanation of variance for each factor is shown in Table II. The first five factors are selected, the first factor (denoted as F1) explained 33.646%, the second factor (denoted as F2) explained 21.988%, the third factor (denoted as F3) explained 20.350%, the fourth factor (denoted as F4) explained 9.961%, and the fifth factor (denoted as F5) explained 9.576%, with a cumulative extractable variance contribution of 95.521%. F1 is named as "Economic Growth Cofactor" with positive direction; F2 is named as "Innovation Cofactor" with

positive direction; F3 is named as "Cofactors of population income and tertiary industry compared with secondary industry", with positive direction; F4 is named as "Cofactor for the number of people in higher education", with a positive direction; F5 is named as "Medical condition cofactor", with a positive direction.

B. Analysis of Enhanced Panel Factor Scores

1) Regional comprehensive scores: Table III is obtained by taking the mean value of the comprehensive scores of each city in the same year as the total evaluation score of the level of synergistic development in the Greater Bay Area over the years, and drawing the corresponding trend line diagram as shown in Fig. 1.

By combining Fig. 1 and Table III, it can be seen that the level of synergistic development (Fsum) of the Greater Bay Area shows an overall upward trend. Among them, the growth is slow around 2008, grows to the average level in 2013, maintains a high level of growth rapidly between 2010 and 2019, and decreases slightly in 2020 due to the epidemic and other factors. It indicates that there is a growing synergy among the Greater Bay Area cities, and the synergistic trend tends to grow steadily.

TABLE III. COMPOSITE SCORE TABLE OF LEVEL OF ECONOMIC SYNERGISTIC DEVELOPMENT OF THE GREATER BAY AREA FROM 2005-2023

Year	F_1	F_2	F_4	F_5	F_{sum}
2005	-0.193	-0.437	0.077	-0.792	-0.864
2006	-0.163	-0.426	0.012	-0.555	-0.699
2007	-0.176	-0.344	-0.001	-0.557	-0.687
2008	-0.144	-0.305	0.055	-0.646	-0.697
2009	-0.210	-0.254	0.095	-0.561	-0.566
2010	-0.112	-0.226	0.089	-0.538	-0.516
2011	-0.032	-0.197	0.069	-0.380	-0.354
2012	0.000	-0.163	0.063	-0.155	-0.122
2013	0.036	-0.113	0.031	0.013	0.032
2014	0.139	-0.135	0.029	0.152	0.182
2015	0.216	-0.034	-0.005	0.318	0.353
2016	0.155	0.189	-0.073	0.523	0.526
2017	0.134	0.341	-0.078	0.648	0.680
2018	0.095	0.583	-0.131	0.778	0.803
2019	0.107	0.699	-0.137	0.896	0.937
2020	0.147	0.821	-0.092	0.855	0.995
2021	0.189	0.943	-0.086	0.924	1.089
2022	0.231	1.065	-0.08	0.993	1.183
2023	0.273	1.187	-0.074	1.062	1.277

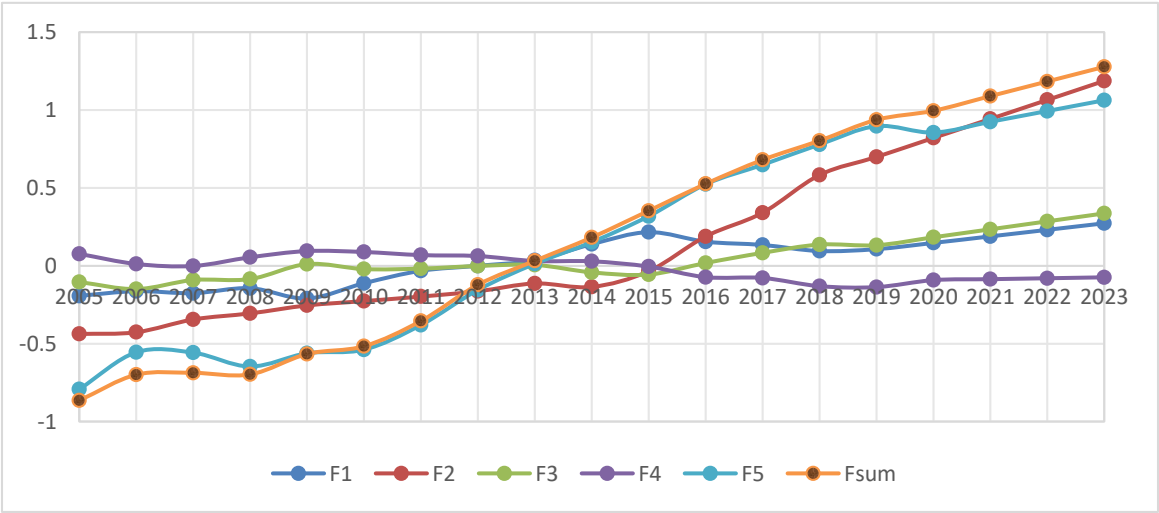


Fig. 1. The trend of level of synergistic development of the Greater Bay Area from 2005-2023.

Economic Growth Cofactor (F1) shows an overall slowly fluctuating upward trend, with the lowest level in 2009, then increasing year by year, reaching the average level in 2012 and the highest level in 2015, and then showing a partial fluctuating downward trend. Innovation Cofactor (F2) shows an overall upward trend, being below average and growing slowly in 2015, but above average and growing faster in 2016 and beyond. Cofactors of population income and tertiary industry compared with secondary industry (F3) show an overall slowly fluctuating upward trend, below average until 2016 (except for 2009 and 2013) and reaching average and slowly increasing trend in 2016 and beyond. Cofactor for the number of people in higher education (F4) shows a small fluctuating overall decreasing trend. Medical condition cofactor (F5) shows an overall upward trend, fluctuating slightly before 2010, increasing faster after 2010, and reaching above average and stable growth in 2013.

The above results, combined with the standard deviation of each factor (results in Table IV), show the changes in each factor. There is an increase and a large change in the F sum and Medical condition cofactor (F5) and Innovation Cofactor (F2); Cofactors of population income and tertiary industry compared with secondary industry (F3) and Economic Growth Cofactor (F1) have increased but not much, indicating that the Greater Bay Area has made relatively small progress in the above two directions; Cofactor for the number of people in higher education (F_4) has decreased but not significantly, indicating a small regression in this direction for the Greater Bay Area.

2) City score: The average comprehensive scores of each city in the Great Bay Area over the years were considered as the overall evaluation scores of their coordinated development levels. The corresponding trend lines are shown in Fig. 2. Three cities have a positive composite score for the level of economic synergistic development, namely Guangzhou, Hong Kong and Zhuhai, with Guangzhou having the highest score for the level of economic synergistic development; the rest of the cities have a negative and below average score. The cities ranked from highest to lowest were Guangzhou, Hong Kong, Zhuhai, Zhongshan, Foshan, Macau, Shenzhen, Dongguan, Jiangmen, Huizhou, and Zhaoqing. In terms of economic growth factor (F1), Hong Kong ranks the highest and Macau the lowest; in terms of Innovation Cofactor (F2), Shenzhen ranked the highest and Hong Kong the lowest; in terms of cofactors of population income and tertiary industry compared with secondary industry (F3), Macau ranks the highest and Foshan the lowest; cofactor for the number of people in higher education (F4); Guangzhou ranks the highest and Shenzhen the lowest; medical condition cofactor (F5), Zhuhai ranks the highest and Zhaoqing the lowest.

Based on the trend graph of scores of Fsum for each city in the Greater Bay Area over the years (see Fig. 3 for details, here the Fsum is used as an example, all other factors can be analyzed similarly), we can see that in general, the level of synergistic development in all cities shows a generally increasing trend year by year, but the levels differ. Guangzhou, Hong Kong, and Zhuhai stand out with higher scores, while other cities show varied growth trends.

TABLE IV. STANDARD DEVIATION OF EACH FACTOR OF ECONOMIC SYNERGY FOR THE GREATER BAY AREA

Factor	F_1	F_2	F_3	F_4	F_5	F_{sum}
Standard Deviation	0.147	0.404	0.093	0.079	0.600	0.650

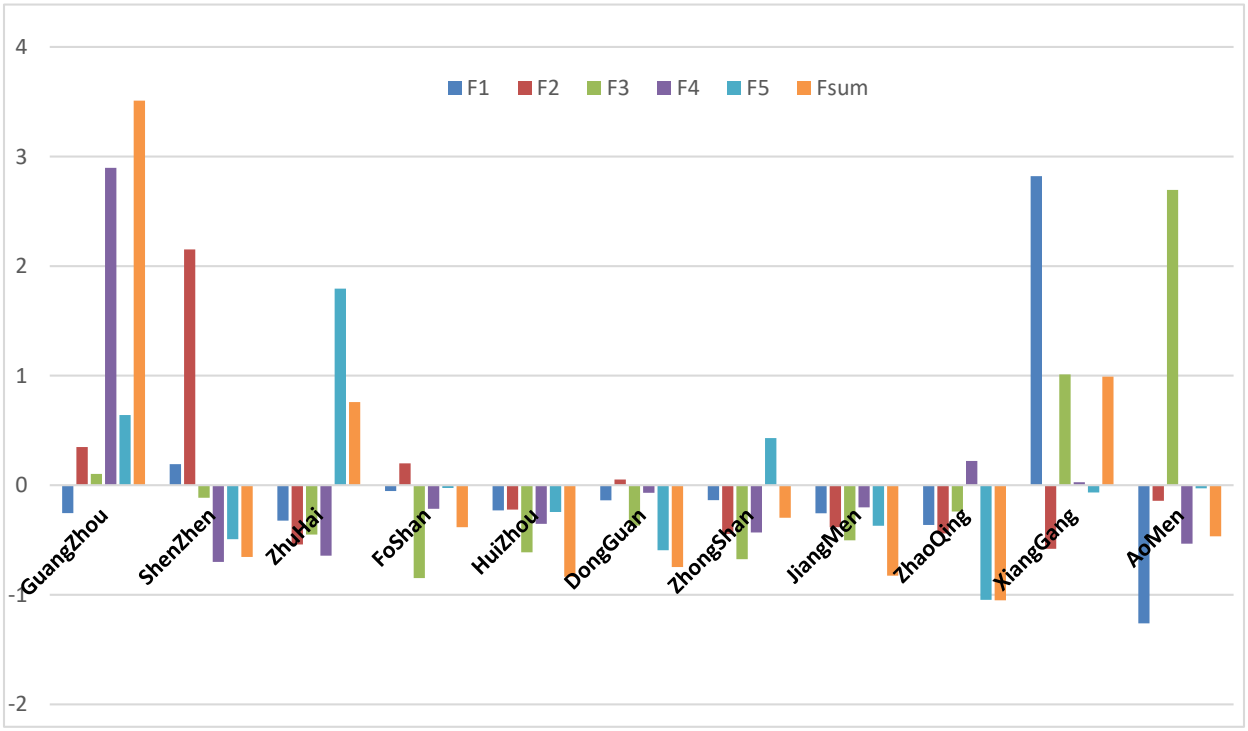


Fig. 2. Comparison of the level of synergistic development by city.

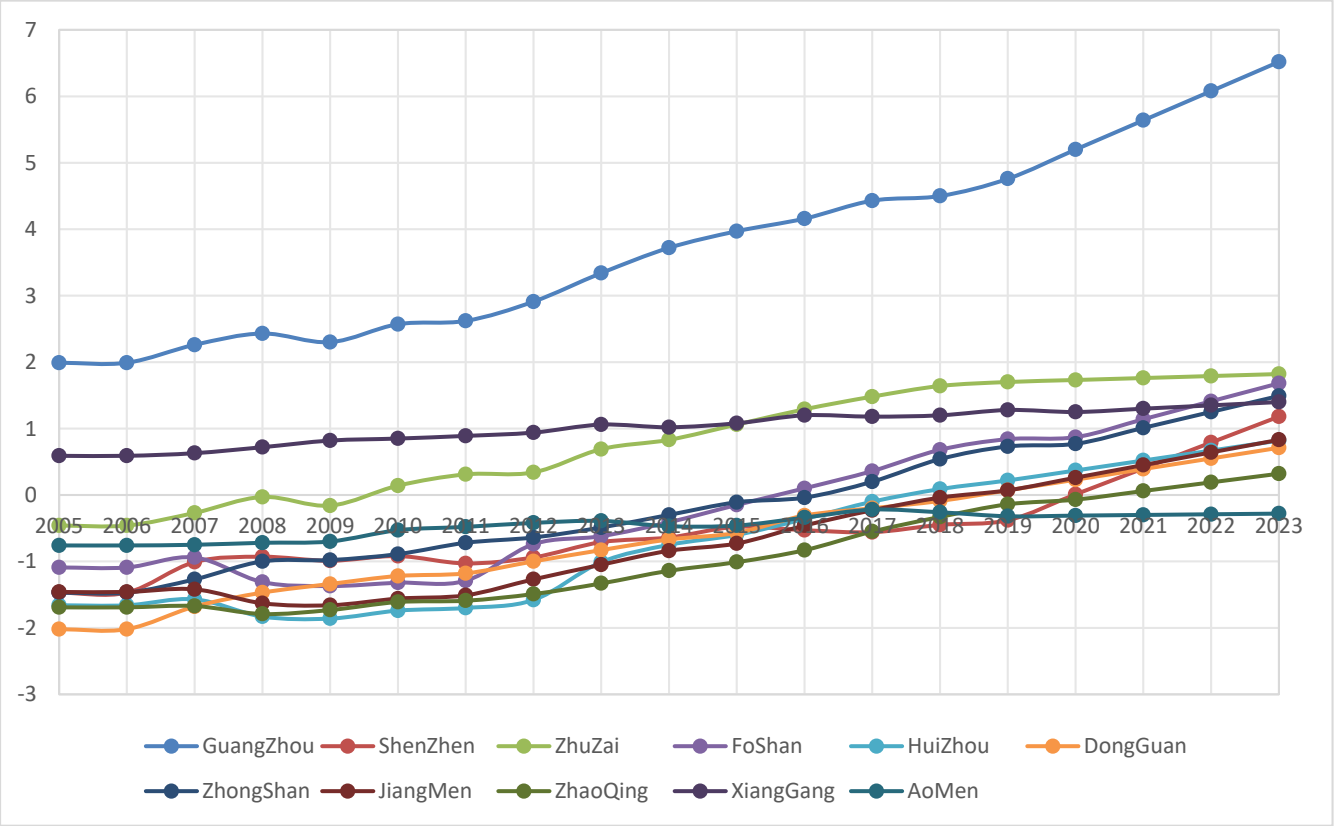


Fig. 3. The composite scores of levels of synergistic development for each city in the greater bay area.

TABLE V. GROWTH RATE AND TOTAL CHANGE OF THE COMPOSITE SCORE OF THE LEVEL OF ECONOMIC SYNERGISTIC DEVELOPMENT OF EACH CITY

City	Growth Rate	No.	Total Change	No.
Guangzhou	0.2256	**1**	3.03	1
Shenzhen	0.0898	9	2.03	6
Zhuhai	0.1656	**3**	2.32	5
Foshan	0.1565	5	1.34	9
Huizhou	0.1796	**2**	2.36	3
Dongguan	0.1491	6	2.36	4
Zhongshan	0.1587	**4**	2.39	2
Jiangmen	0.1442	7	1.77	8
Zhaoqing	0.1355	8	1.85	7
Hong Kong	0.0448	10	0.57	10
Macau	0.0339	11	0.42	11

Note: Growth rate = slope of the regression model with calendar year scores as the dependent variable and time as the independent variable; total change = $F_{sum\ 2023}$ - $F_{sum\ 2005}$.

By calculating the standard deviation and total change of the level of synergistic development of each city over the years (see Table V), it can be seen that the top three cities with the largest growth rate of the level of synergistic development are Guangzhou, Huizhou and Zhuhai, the smallest are Macau, Hong Kong and Shenzhen, while Zhongshan, Foshan, Dongguan, Jiangmen and Zhaoqing are in the middle level. To conclude, in the process of synergistic development of Guangdong, Hong Kong and Macau, the nine mainland cities have changed more than Hong Kong and Macau, and Hong Kong and Macau have a greater synergistic effect on the nine mainland cities.

C. *Evaluation Results of Machine Learning Regression Models*

To further verify the scientific validity and accuracy of the comprehensive score Fsum obtained from the factor analysis method and its factor structure, and to enhance the reliability of the research results, this paper introduces two machine learning models - multiple linear regression and LASSO regression - to empirically analyze the relationship between the main feature variables and Fsum. The model evaluation results are as follows:

As shown in Tables VI and VII, both regression models achieve high goodness-of-fit, indicating strong explanatory power of the feature variables for the comprehensive factor score (Fsum). The linear regression model reveals that the number of enrolled university students, hospital beds per 10,000 people, and the proportion of tertiary industry value added are core positive influencing factors of Fsum, with some macroeconomic variables showing statistical significance. Other variables have limited impact. The LASSO regression model automatically selects a small number of highly explanatory variables, further highlighting the key roles of education and healthcare resources and industrial structure on Fsum. The conclusions from both models are highly consistent and mutually validating, enhancing the scientific rigor and robustness of the study results.

TABLE VI. COMPARISON OF OVERALL GOODNESS-OF-FIT FOR REGRESSION MODELS

Model Type	R^2	Adjusted R^2	Mean Squared Error (MSE)
Linear Regression	0.81	0.79	0.41
LASSO Regression	0.85	0.81	0.38

TABLE VII. COMPARISON OF REGRESSION COEFFICIENTS AND THEIR SIGNIFICANCE FOR KEY FEATURES

Variable Name	Linear Regression Coefficient	Linear Regression p-value	LASSO Regression Coefficient	Variable Effect Description
(Intercept)	-5.30	<0.001***	0	Intercept
Number of University Students	0.0000034	<0.001***	0.77	Major positive influencing factor
Hospital Beds per 10,000 People	0.044	<0.001***	0.54	Major positive influencing factor
Share of Tertiary Industry (%)	0.044	<0.001***	0.09	Major positive influencing factor
Total GDP (RMB 100 million)	-0.00013	0.006**	0	Significant in linear regression, shrunk to zero in LASSO
Share of Secondary Industry (%)	0.033	0.002**	0	Significant in linear regression, shrunk to zero in LASSO

Note: *p<0.05, **p<0.01, ***p<0.001. A LASSO coefficient of 0 indicates that the variable was excluded by the model.

V. CONCLUSIONS

This study integrates a panel factor model with machine learning algorithms to assess the level and structural determinants of regional collaborative development in the Guangdong-Hong Kong-Macao Greater Bay Area. The results demonstrate that the factor model effectively synthesizes redundant information, isolates key latent factors, and achieves high explanatory power, with the top three factors accounting for 76% and the top five accounting for 96% of the total variance. Linear regression confirms significant relationships between factors and collaborative development, whereas LASSO regression exhibits superior robustness and sparsity in high-dimensional contexts. Consistency across methods strengthens the scientific validity of the findings, which indicate steady overall progress but persistent regional disparities, suggesting scope for further optimization.

Theoretically, this work demonstrates the value of combining traditional econometric models, characterized by strong theoretical foundations, interpretability, and stability, with machine learning methods, which excel in feature selection, pattern recognition, and predictive accuracy. This integration unites explanatory depth with predictive precision, and offers a more rigorous and robust framework for regional development assessment. Practically, it highlights the importance of addressing both structural drivers and dynamic prediction in strategy design, enabling the identification of industrial, talent, and technological levers, as well as the early detection of disparities and bottlenecks.

The methodological framework presented here is transferable to other economically significant city clusters, such as the Beijing-Tianjin-Hebei region, the Yangtze River Delta, the Tokyo Bay Area, and the London metropolitan area. Beyond regional studies, it can be applied to domains such as real estate market health assessment, industrial chain resilience, energy structure optimization, transportation network efficiency, technological innovation capacity, and cross-regional economic integration.

The machine learning models employed in this study are relatively limited in scope, and the findings may therefore be subject to certain constraints. Future research will broaden methodological diversity and delve more deeply into the integrated application of econometric and machine learning approaches to complex socio-economic development challenges.

ACKNOWLEDGMENT

The authors gratefully acknowledge the intelligent AI assistant Kimi for its valuable assistance in translating several technical terms during the preparation of this manuscript.

The authors would also like to express sincere gratitude to the Guangzhou Huashang College National Scientific Research Project Team (2024HSKT02), the Construction Project of the Core Curriculum Teaching and Research Office for Financial Technology, Guangzhou Huashang College (HS2024ZLGC43), and the Research Project on the Mechanism Innovation and Implementation Path of High-Quality Development of SMEs throughout the Life Cycle Driven by Fintech in the Guangdong-Hong Kong-Macao Greater Bay Area (No. 2022ZDJS125) for their valuable contributions in topic analysis and journal recommendations.

REFERENCES

[1] Zhang M. Y., Quan R. 2020. Evaluation of coordinated development in the Beijing-Tianjin-Hebei region. Statistics and Decision, 36(04), 72–76. (In Chinese).

[2] Shi X. T., Yu Z. Y. 2025. Measurement of the new quality productivity level, regional differences, and synergy of the composite system. Science & Technology Progress and Policy, 42(07), 9–20. (In Chinese).

[3] Li J. M., Yin Q., Zhang Y. Z., Qi J. J. 2023. Synergistic effect analysis of "open corridor-border port-border regional economy" composite system. Commercial Economy Research, (05), 171–175. (In Chinese).

[4] Xiong X. L., Fan J. 2022. Coordinated evolution mechanism and regional coordination differences between fintech and green transformation of manufacturing: Empirical analysis based on the Haken model. Science & Technology Progress and Policy, 39(11), 63–73. (In Chinese).

[5] Huang G. L., Xu R. Y., Wang X. Z., Li M. B. 2020. Evaluation of regional innovation synergy in Henan Province based on the composite system synergy model. Journal of Henan Agricultural University, 54(06), 1059–1066. (In Chinese).

[6] Ma X. 2019. Evaluation of economic synergy in the Beijing-Tianjin-Hebei region based on the composite system synergy model. Industrial Technology Economy, 38(05), 121–126. (In Chinese).

[7] Chen Y., Miao Q. Q., Zhou Q. 2022. Spatiotemporal differentiation and driving force analysis of the high-quality development of urban agglomerations along the Yellow River Basin. International Journal of Environmental Research and Public Health, 19, 2484.

[8] Yi P., Dong Q., Li W., et al. 2022. Assessment of city sustainability with the consideration of synergy among economy–society–environment criteria. Environment, Development and Sustainability. https://doi.org/10.1007/s10668-022-02364-w

[9] Zhang P., Liu Z., Liu X., et al. 2021. Does functional specialization affect collaborative development? Evidence from four Chinese metropolitan areas. Environmental Science and Pollution Research, 28, 41851–41868.

[10] Ding R. Z., Zhao X. Z. 2025. Measurement of industrial synergy in China's metropolitan areas. Journal of Tongji University (Social Science Edition), 36(03), 89–100. (In Chinese).

[11] Zhou J. K., Han L., Zhang X. Y., et al. 2025. Industrial coordinated development in the Beijing-Tianjin-Hebei region: Model, performance, and policy choices. Economic Geography, 45(03), 238–245. (In Chinese).

[12] Sun T. S. 2025. Industrial diversification and regional economic resilience in Beijing-Tianjin-Hebei: Based on the perspective of intra- and inter-regional industrial related diversity. Tianjin Social Sciences, (01), 107–116. (In Chinese).

[13] Qin P., Zhang Y. J., Xiao Q. Y. 2025. Research on the cross-regional coordination mechanism of innovation chain and industrial chain in Beijing-Tianjin-Hebei—Spatio-temporal evolution analysis based on strategic emerging industries. Journal of Beijing Union University (Humanities and Social Sciences Edition), 23(04), 1–14. (In Chinese).

[14] Lu J., Guo Z. A., Wang Y. P. 2024. Development level, regional differences, and promotion path of new quality productivity. Journal of Chongqing University (Social Science Edition), 30(03), 1–17. (In Chinese).

[15] Yan L. J., Zhao Y. J., Qiu S. K., et al. 2022. Construction and evaluation of high-quality development indicator system in the Yellow River Basin—Taking Henan section as an example. Regional Research and Development, 41(06), 37–43. (In Chinese).

[16] Xue W., Cai C. 2022. Comprehensive evaluation of high-quality development in China based on multilevel factor analysis. Statistics and Decision, 38(18), 22–25. (In Chinese).

[17] Li F. S., Ci F. Y. 2025. Spatiotemporal evolution and driving factors of coupling coordination among China's digital economy, carbon emissions efficiency, and high-quality economic development. Sustainability, 17(14), 6410.

[18] Chang S. X., Xue Q., Xu S. Y. 2023. Dynamic evaluation and comparison of the coordinated development level in the Beijing-Tianjin-Hebei region based on the entropy method. Industrial Innovation Research, (15), 22–24. (In Chinese).

[19] Ye Y. Y., Wang X. Y., Xu J. L., et al. 2022. Connotation and mechanism changes of coordinated development in the Guangdong-Hong Kong-Macao Greater Bay Area in the new era. Tropical Geography, 42(02), 161–170. (In Chinese).

[20] Gao J. Y. 2022. Evaluation of industrial policy effects and industrial synergy degree in the Guangdong-Hong Kong-Macao Greater Bay Area. Science and Technology Management Research, 42(12), 53–58. (In Chinese).

[21] Li Y. C., Peng W. K., Yang Y. H. 2024. Research on urban functional differences and guiding strategies in the Guangdong-Hong Kong-Macao Greater Bay Area: An analysis based on the technological innovation network perspective. Urban Problems, 2024(11), 54–62. DOI:10.13239/j.bjsshkxy.cswt.241107. (In Chinese).

[22] Wang S. B., Qin Y., Lin X., et al. 2025. Network structure and synergy characteristics in the Guangdong-Hong Kong-Macao Greater Bay Area. Applied Sciences, 15(14), 7705. https://doi.org/10.3390/app15147705

[23] Lin H. M. 2006. The exact model of factor analysis and its solution. Statistics & Decision, 2006(14), 4–5. (In Chinese).

[24] Lin H. M. 2009. Improvement and application of factor analysis model. Journal of Applied Statistics and Management, 28(06), 998–1012. (In Chinese).

[25] Zhao H. Q., Zhu J. P. 2019. How to calculate factor analysis application results with SPSS software. Statistics & Decision, 35(20), 72–77. (In Chinese).

[26] Shi L., Lin H. M. 2020. Statistical evaluation and empirical evidence of the degree of economic openness—Guangdong Province as an example. Statistics & Decision, 36(24), 89–93. (In Chinese).

CrypTen-FL: A Secure Federated Learning Framework for Multi-Disease Prediction from MIMIC-IV Using Encrypted EHRs

Himanshu, Pushpendra Singh
Department of Computer Science and Engineering, SRM Institute of Science and Technology,
Delhi-NCR Campus, Ghaziabad, Uttar Pradesh, India-201204

***Abstract*—The increasing demand for privacy-preserving machine learning in healthcare has driven the need for federated approaches that ensure data confidentiality across institutions. In this work, we present CrypTen-FL, a secure federated learning framework for disease prediction using the MIMIC-IV electronic health record (EHR) dataset. CrypTen-FL enables collaborative model training across multiple hospitals without sharing raw patient data, thereby addressing critical privacy concerns through the integration of Secure Multi-Party Computation (SMPC) using CrypTen and differential privacy mechanisms. We adopt a Transformer-based neural architecture to effectively capture the temporal and high-dimensional nature of EHR data, enabling accurate prediction of multiple clinically significant conditions. The framework incorporates decentralized key generation, secure aggregation, and cross-institutional evaluation to assess generalization performance and robustness. Experimental results demonstrate that CrypTen-FL achieves competitive predictive performance while offering strong privacy guarantees, paving the way for secure and scalable AI applications in real-world healthcare settings.**

***Keywords*—Federated learning; secure multi-party computation; electronic health records; disease prediction; MIMIC-IV**

I. Introduction

The use of machine learning in healthcare has created important opportunities for improvements in diagnosis, prognosis, and personalized treatment planning for diseases [1]. Data-driven models are now a reliable way of predicting clinical outcomes for diverse illnesses, as electronic health records (EHRs) are widely available [2], [3]. The traditional approach of bringing together patient data for model development raises significant data security [5], privacy [4], and legal challenges involving the Health Insurance Portability and Accountability Act (HIPAA) and the General Data Protection Regulation (GDPR). These aspects [6] become even more salient in multi-institutional settings [7], since patient data is split among hospitals with various access and privacy rules and restrictions that complicate collaborative, efficient development of machine learning models that are widely relevant.

Numerous machine learning (ML) [8], [9], [10], and deep learning (DL) [11], [12], and [13] methods are helping with early diagnosis and treatment options thanks to recent applications in the medical area. They can recognize trends across patient data to predict risk for various diseases, enabled by use of large electronic health records (EMRs) [14], [15]. Because of the limitations of single-condition approaches [16], the field has moved toward multiple diseases prediction [17],[18]. This captures solutions to understanding many diseases at once, while real-world patients often suffer from multiple diseases. The models [19] can look at multiple burdens of disease at once by assessing complex clinical data to create a stratified intervention for personalized treatment.

Patient health information presents privacy issues because it is sensitive [20], [21], especially in machine learning systems with healthcare focuses, like with federated learning [22]. Electronic health records (EHRs) are a rich yet very private source of data that can include laboratory results, medical histories, documenting diagnoses and treatments, and personally identifiable information (PII) [23]. Unauthorized access and/or exposure to accessing such information can create serious issues–including identity theft, insurance discrimination, legal liability, and decreased patient trust in healthcare organizations. Healthcare data is also typically protected under legal rules, which enforce strict requirements for data access, data sharing, and safeguards in place for privacy and confidentiality; as examples, this includes HIPAA in the U.S. and the GDPR in the EU [24]. While raw data is not shared in federated learning systems, vulnerabilities such as model inversion or gradient leaking could expose private information during model upgradesThus, robust privacy-preserving techniques [25] are required to ensure that private information is not accidentally disclosed during collaborative model training [26] and to enable the safe and lawful application of AI [27], [28] in healthcare environments.

Federated Learning (FL) [29], [30] is a viable paradigm to address these problems because it permits the collaborative training of machine learning models across decentralized data silos without requiring direct access to sensitive patient data. In the Federated Learning (FL) scenario described in [31], the institutions keep their local datasets, while only sharing model updates or encrypted parameters with a central (coordinator) and possibly with other participating peers. This decentralized approach maintains data localization, while reducing the risks of data centralization. Even with their intrinsic privacy properties, standard FL frameworks are vulnerable to many types of privacy leaks, including membership inference and gradient inversion attacks. These vulnerabilities pose a significant hurdle to deploying FL systems in sensitive domains, like healthcare, where even a small amount of information leakage can result in

serious ethical and legal consequences [32], [33]. Hence, FL methods that are even more privacy preserving [34] are needed for FL to be viable and trusted in a clinical context.

In conclusion, designing secure and private machine learning systems for illness prediction on EHR-based data involves a comprehensive approach incorporating federated model architectures, privacy-aware learning algorithms, and cryptographic protocols. These issues must be addressed, as trust, regulatory compliance, and the safe application of AI in clinical settings depend on it. While sensitive data—banking, medical, or personal records—are pivotal in developing machine learning applications, they also pose significant privacy issues. Due to the risks of data leakage and misuse inherent in centralized approaches, there is a significant need for solutions that are both secure and privacy-preserving. CrypTen presents a viable framework to address these concerns, utilizing secure multiparty computation to allow collaborative model training using raw data. This motivates a need to balance data secrecy with the utilization of machine learning's capabilities. Sections I and II provide background on secure computation, federated learning and privacy-preserving machine learning. Section III describes the materials, methods, datasets, and study protocols used in the research. This section, and beyond, focuses on the methods and model architecture detailing how CrypTen facilitated secure learning. Results, analysis, and case studies are covered in Section IV in order to assess performance and usefulness. Lastly, we wrap up the work by identifying its shortcomings, summarizing its contributions, and suggesting areas for further investigation.

II. Background

Khaled et al. [35], Early detection systems, individualized treatments, and AI-driven clinical tools have all been made possible by the MIMIC datasets, which offer de-identified ICU patient data. Strong performance in analyzing irregular vital signs has been demonstrated using convolutional deep learning models. Demographic bias, poor data quality, limited model generalizability, and reproducibility problems, such as more than 25% sample size variations in replication studies remain obstacles, nevertheless. Using MIMIC data, this study highlights important approaches and enduring difficulties in critical care research.

Qian et al. [36], in this study, they discovered that as ED-to-ICU transfer periods increased, in-hospital mortality decreased from 17.6% to 12.2%, with a median delay of 3.98 hours associated with noticeably lower mortality. Even without corrections, the risks of mortality were 25% lower for patients in the longest delay quartile (Q4) than for those in the lowest quartile (Q1). Additionally, shorter ICU stays were linked to longer ED stays. These findings underline the need of improving ED care and transfer procedures by indicating that prolonged ED stay may enable crucial stabilization.

Damme et al. [37], they reported a significant improvement After converting the MIMIC-ED dataset to the FHIR format, its interoperability and reusability were improved, and its FAIRness score increased from 60 to 82 out of 95. Although FHIR increases the accessibility and reusability of data, clear and thorough implementation standards are necessary for a successful FAIR implementation. Adopting FHIR enhances dataset interoperability and promotes wider community harmonization through standardized processes, as the MIMIC-ED case study illustrates.

Wang et al. [38], in the context of SAPS III fared better than other widely used scoring systems, such as GAS, SAPS II, SOFA, SIRS, and OASIS, in predicting 28-day and 1-year mortality among intensive care unit (ICU) patients with non-ruptured abdominal aortic aneurysms (AA). The study reported an AUROC of 0.805 for SAPS III, identifying it as the most accurate predictor and an independent risk factor for mortality. While GAS has traditionally been used for ruptured AA, its performance in this subgroup was comparatively lower. These findings underscore the superior prognostic utility of SAPS III in critically ill AA patients undergoing both endovascular and surgical interventions.

Sadeghi et al. [39], did a comparison on SICdb and MIMIC-IV showed that SICdb offers higher temporal resolution and more frequent vital sign data, making it better suited for longitudinal studies. While SICdb provides detailed physiological signals and European healthcare data for AI benchmarking, it lacks clinical notes, imaging, and secondary diagnoses. MIMIC-IV, though less granular, offers broader clinical coverage.

Wang and Jin et al. [40], used machine learning to predict prostate cancer risk, with LightGBM outperforming other models (AUC 0.93, sensitivity 86%, specificity 85%). Key risk factors identified were age, renal disease, and platelet count. Using MIMIC-IV data (11,745 BPH and 1,975 cancer cases), the study favored LightGBM over deep learning due to challenges with tabular data and limited sample size.

Horvath et al. [41], shows that FedProx outperforms FedAvg on MIMIC-III under data heterogeneity. Despite some privacy leakage, DP-SGD and DP-SVT maintain performance close to non-private models. Larger gradient norms help balance privacy and accuracy, highlighting trade-offs in federated learning with differential privacy.

Sun et al. [43], developed a nomogram using LASSO to predict in-hospital mortality (53.95%) in ICU cardiac arrest patients, achieving the highest performance (AUC 0.79) among tested models. Key predictors included SAPS III, bicarbonate, PT, and NEWS 2, making it a valuable tool for clinical decision-making.

Kaminaga et al. [44], proposed MPCFL architecture combines Federated Learning with Multi-Party Computation to enable secure, collaborative fraud detection without exposing data or model updates. Tested on different datasets, it effectively prevents common FL threats while maintaining model accuracy, proving its potential for privacy-preserving machine learning.

Knott et al. [45] presents CRYPTEN, an MPC framework enabling privacy-preserving training and inference via encrypted tensors. Tested on models like ResNet-18 and ViT-B/16, it achieves significant GPU speedups but faces high communication costs and debugging challenge.

Budrionis et al. [46], finds that federated learning achieves similar accuracy to centralized training but with significant overhead up to 9× longer training and 40× longer inference

times. Using PySyft with Docker, it shows federated setups are resource-intensive yet enable privacy-preserving, regulation-compliant learning on real-world healthcare data.

Kanagavelu et al. [47], proposes a two-phase MPC-enabled FL framework for Industrial IoT, improving scalability and reducing communication overhead. Additive MPC proves more efficient than Shamir sharing, achieving 2–25× faster execution while preserving accuracy and privacy. Future work aims to scale across regions and enhance security against malicious threats.

Zhu et al. [48], formalizes Federated Learning (FL) as a subset of Secure Multi-Party Computation (SMPC), modeling each training round as a secure m-ary function. Using a simulation-based security framework, it expresses the overall FL process as a composition of round-wise computations.

Byrd et al. [49], in this paper presents a privacy-preserving federated logistic regression protocol for fraud detection, combining MPC to protect client model weights and DP to prevent inference attacks. Tested on real-world credit card data, the method shows strong scalability and maintains accuracy with increased privacy (lower ε) and more participants.

Truex et al. [50] presents a federated learning system combining Secure Multi-Party Computation and Differential Privacy to ensure strong privacy with high accuracy. Tailored to a specific threat model, it achieves F1-scores above 0.87 up to 0.9 in some cases, outperforming existing private FL methods in both scalability and performance. It supports multiple ML algorithms, proving its practicality for privacy-sensitive tasks.

III. Material and Methods

A. Database

We utilize the MIMIC-IV dataset [42] to develop and evaluate our federated disease prediction framework. MIMIC-IV contains de-identified, structured clinical data from ICU patients, including demographics, vitals, laboratory results, medications, and diagnostic codes. For our model, we focus on extracting time-series features (e.g. heart rate, blood pressure, creatinine, WBC count) along with static attributes (e.g. age, sex, ethnicity) to serve as input to a deep LSTM-based neural network. These features are temporally aligned and normalized per patient to construct multivariate sequences suitable for temporal modeling. The dataset is partitioned across multiple simulated hospital nodes, each representing a federated learning client with a unique subset of patient records. This partitioning is designed to emulate real-world inter-hospital data silos and enables training under both IID and non-IID conditions. Local preprocessing, including missing value imputation, sequence padding, and categorical encoding, is performed independently at each client. No raw data is shared across institutions; instead, model updates are trained locally and encrypted using CrypTen's SMPC before being sent for secure aggregation. This data partitioning and local preprocessing pipeline is critical for maintaining the decentralized and privacy-preserving nature of our federated model.

B. Experimental Setup

To evaluate the proposed CrypTen-FL framework, we conducted experiments on twelve clinically significant diseases using the MIMIC-IV dataset. Each disease ranging from heart failure and sepsis to AKI and neurological disorders was framed as a binary classification task. Patient data (demographics, vitals, labs, and medications) was preprocessed and partitioned across four virtual hospitals to simulate real-world cross-institutional data silos. We implemented five federated learning strategies: Horizontal FL (FedAvg), Vertical FL, Personalized FL (FedPer), Multi-task FL (FedMTL), and Split-FL. All training was conducted using the CrypTen Secure Multi-Party Computation (SMPC) environment to preserve privacy through encrypted model updates and secure aggregation. Training ran for 20 communication rounds using the Adam optimizer (learning rate = 0.001), batch size 128, and binary cross-entropy loss. Additionally, differential privacy was optionally enforced via DP-SGD with a privacy budget of $\varepsilon = 3.0$. Model performance was assessed using encrypted ROC-AUC, accuracy, precision, recall, and F1-score, securely computed using binning techniques to prevent information leakage.

C. Evaluation

To rigorously assess disease prediction performance across hospitals using the MIMIC-IV dataset, we employed a suite of standard classification metrics: Accuracy, Precision, Recall, F1-score, Specificity, ROC-AUC, and Mean Squared Error (MSE). These metrics provide a comprehensive understanding of the model's predictive ability, calibration, and fairness in a multi-institutional, heterogeneous healthcare environment.

In the context of MIMIC-IV, which contains rich, high-dimensional clinical data from multiple hospitals and diverse patient populations, it is crucial to evaluate models on both their discriminatory power and their fairness across subgroups. Federated models trained with CrypTen's Secure Multi-Party Computation (SMPC) allowed these evaluations to be conducted without sharing sensitive patient-level data, ensuring compliance with privacy standards (e.g. HIPAA, GDPR). Let the confusion matrix elements be: True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN) and N is the total number of patients evaluated. To determine the proportion of total correct predictions across all patient cases is calculated by Eq. (1).

$$Accuracy = TP + TN/(TP + TN + FP + FN) \quad (1)$$

Precision is used to determine the ability of the model to correctly identify positive disease cases among those predicted positive by Eq. (2).

$$Precision = TP/(TP + FP) \quad (2)$$

Recall defined the ability to detect actual positive disease cases in the dataset defined in Eq. (3).

$$Recall = TP/(TP + FP) \quad (3)$$

To determine the balance between precision and recall, particularly important given class imbalances (e.g., rare disease cases), F-1 score by Eq. (4).

$$F1 - Score = 2 \times (Precision \times Recall)/(Precision + Recall \quad (4)$$

Specificity is the effectiveness in correctly identifying patients without the disease by Eq. (5).

$$Specificity = TN/(TN + FP) \quad (5)$$

ROC-AUC which captures the model's ability to distinguish between disease and non-disease cases across thresholds. Calculated by integrating the ROC curve (true positive rate vs. false positive rate). Eq. (6), the Mean Squared Error (MSE) calculates the squared difference between projected probability and actual illness labels.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}i - yi)^2 \quad (6)$$

This comprehensive metric suite ensures reliable evaluation of federated illness prediction models trained on MIMIC-IV by safeguarding privacy and promoting equity across hospital sites and patient subgroups.

D. Methodology

This section describes preprocessing, model architecture, Secure Multi-Party Computation (SMPC), FL.

1) Preprocessing: We drew upon MIMIC-IV, a large, de-identified electronic health record (EHR) repository that is freely available to the research community and contains data from over 380,000 hospital admissions. The primary aim of the study was to predict clinically meaningful conditions/disorders, including diabetes mellitus, sepsis, heart failure, pneumonia, chronic kidney disease (CKD), and urinary tract infections (UTIs). Within the codebook, we identified continuous variables, categorical features, and temporal records, which included structured data items (e.g. vital signs, demographics, laboratory results, diagnoses [ICD-10 codes], prescriptions, and clinical procedures). To simulate a federated healthcare system, the data were split into non-overlapping arbitrary groupings which illustrated hospital silos within continuous variable cohorts, categorical features, matched temporal records by admission ID, and used imputation of categorical and continuous missing data, employing a mix of statistical and clinical heuristics. The final cohort of patients excluded any patients with missing disease designations and inadequate temporal records.

2) Federated learning configurations: In Horizontal Federated Learning (HFL), The datasets from collaborating institutions have distinct patient populations but the same feature space (e.g. clinical factors). In order to provide collaborative learning without direct data exchange, each institution trains its model locally using its own patient information. Only encrypted model updates are sent for secure aggregation.

Vertical Federated Learning (VFL), is appropriate, on the other hand, when organizations maintain different feature sets but share overlapping patient populations (e.g. labs in one hospital and imaging data in another). In order to facilitate collaborative model optimization while preserving data confidentiality, the training procedure in this case uses secure entity alignment and encrypted feature exchange between sites.

Federated Transfer Learning (FTL) is used when different institutions have different patient groups and feature spaces. By aligning representations in a common latent space, FTL uses domain adaptation techniques to transmit knowledge across clients, enabling learning in extremely diverse environments.

Split Learning (SplitFL) divides the neural network's client and server components. Clients do forward propagation on the initial network tiers using local data. A coordinating server then receives the intermediate feature maps and completes the remaining computations. Because the entire model architecture and source data are maintained decentralized, this setup allows for collaborative training with minimal risk of data leaks.

3) Secure federated learning with SMPC and differential privacy: Secure Multi-Party Computation (SMPC) using CrypTen is used by the CrypTen-FL framework to ensure safe federated learning. Shamir's Secret Sharing enables decentralized key generation, where no single party has complete control. To ensure that no raw data or intermediate findings are ever made public, all computations—including training processes and evaluation measures like ROC-AUC—are conducted entirely in encrypted space. The gradients and model parameters are separated into additive secret shares.

The trade-off between system cost and privacy is guided by a Quant Mapping Reference, which enables dynamic setup according to institutional limitations. While related overheads range from Low (0–30%) to Very High (81–100%), privacy levels range from Low (0.4) to Very High (1.0). The deployment of CrypTen-FL in a variety of clinical settings with differing resource capacities is made possible by this adaptable privacy tailoring.

4) Model architecture: As seen in Fig. 1, we successfully modeled the high dimensionality and temporal dynamics of electronic health record (EHR) data using a Transformer-based neural architecture. This approach handles a variety of clinical variables across time while capturing sequential patterns in patient histories.

A structured time-series format is created from each patient's electronic health record (EHR), with each time step recording a fixed-interval snapshot (e.g. every 6 or 24 hours) of clinical data such vital signs, lab results, prescription drugs, diagnoses, and demographics.

The sequence can be aggregated into a fixed-length embedding vector using either global average pooling or a [CLS] token. This patient-level embedding captures the temporal and multivariate nature of MIMIC-IV data. It is then passed through fully connected dense layers with ReLU activations and dropout for regularization, followed by a sigmoid activation layer. The sigmoid function enables multi-label disease prediction—crucial for MIMIC-IV cases, where patients often exhibit multiple co-occurring conditions such as sepsis, AKI, or heart failure—making the model suitable for real-world ICU applications.

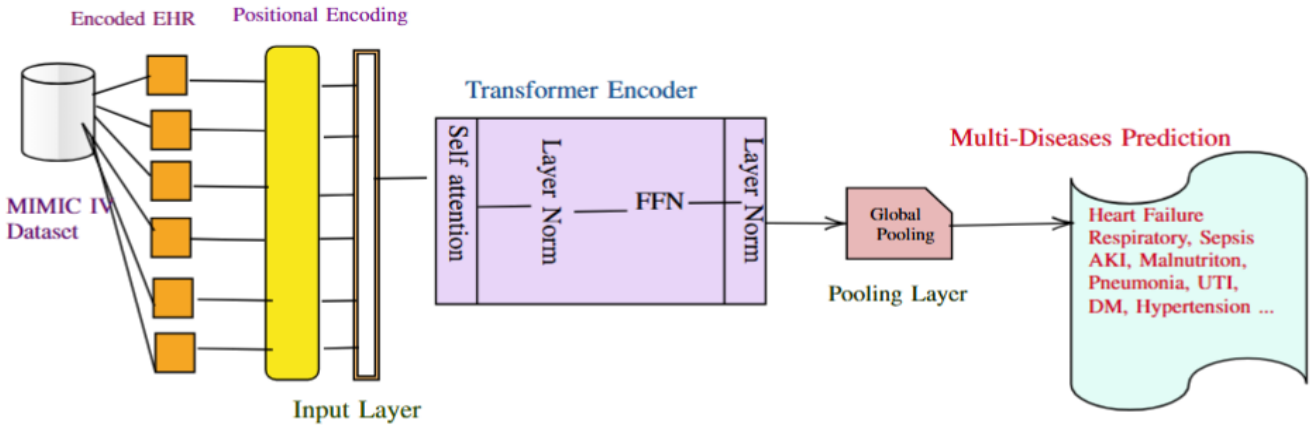


Fig. 1. Model architecture.

IV. Results Analysis

A. Multi-Disease Prediction Performance

The CrypTen-FL framework was evaluated for 12 critical diseases across ICU patients in the MIMIC-IV dataset using non-IID, hospital-partitioned data. The model utilized CrypTen's SMPC backend to ensure secure aggregation of encrypted model updates.

The performance evaluation of the CrypTen-FL framework, as shown in Table I across 12 clinically significant diseases on the MIMIC-IV dataset, demonstrates consistent and robust results across multiple classification metrics. As shown in the table, key performance indicators—ROC-AUC, Accuracy, Precision, Recall, and F1-Score—remain above 0.75 for all disease categories, indicating reliable predictive capability. High ROC-AUC values (ranging from 0.80 to 0.88) across diseases such as sepsis, chronic kidney disease, and heart failure indicate strong model discriminability in distinguishing positive and negative cases, even under secure federated settings with privacy-preserving constraints.

Diseases, including sepsis, heart failure, and chronic renal disease, have scores between 0.82 and 0.84, indicating both good sensitivity and specificity. F1-Scores, which strike a balance between precision and recall, also demonstrate consistent performance overall. Conditions include liver illness and urinary tract infections may have relatively low scores, which could indicate issues with class imbalance or more subtle clinical patterns.

TABLE I. Evaluation of Federated Disease Prediction Models Across 12 Conditions Using Standard Performance Metrics, with Notably Strong Results for Sepsis, CKD, and Heart Failure

Disease	ROC-AUC	Accuracy	Precision	Recall	F1-Score	Specificity
Heart Failure	0.87	0.83	0.80	0.84	0.82	0.82
Respiratory Failure	0.85	0.81	0.77	0.82	0.79	0.80
Sepsis	0.92	0.94	0.83	0.85	0.84	0.83
AKI	0.86	0.82	0.79	0.83	0.81	0.81
Malnutrition	0.82	0.78	0.74	0.77	0.75	0.76
Pneumonia	0.84	0.80	0.76	0.79	0.77	0.78
UTI	0.81	0.78	0.73	0.76	0.74	0.75
Diabetes Mellitus	0.86	0.82	0.79	0.82	0.80	0.81
Hypertension	0.85	0.81	0.77	0.80	0.78	0.79
CKD	0.87	0.83	0.81	0.84	0.82	0.82
Liver Disease	0.80	0.76	0.71	0.75	0.73	0.74
Neurological Disease	0.83	0.79	0.75	0.78	0.76	0.77

B. Performance of FL Models Across Diseases

We examined the performance of a variety of Federated Learning (FL) strategies for disease prediction modeled using the MIMIC-IV dataset. Five different Federated Learning strategies were tried: Horizontal FL, Vertical FL, Personalized FL, Multi-task FL, and Split-FL. We employed these approaches on twelve clinical diseases of significance including heart failure, respiratory failure, sepsis, AKI malnutrition, pneumonia, UTI, diabetes, hypertension, CKD, liver disease, and neurological disorders.

Overall, the comparative ROC-AUC scores shown in Fig. 2 indicate that Multi-task FL obtained the best predictive performance across most disease types taking advantage of shared representations for co-morbid conditions. Personalized FL was not too far behind and was able to effectively adapt across hospitals to non-IID distributions, while maintaining consistently high ROC-AUC values.

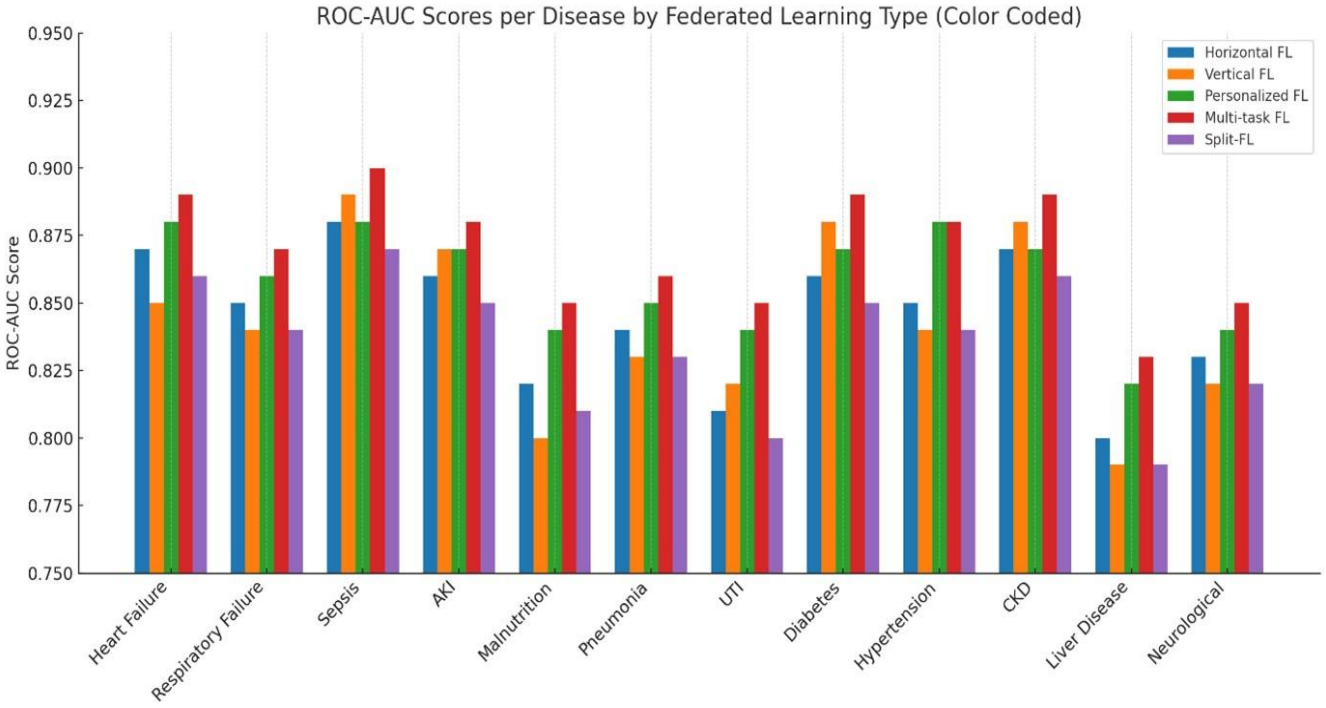


Fig. 2. ROC-AUC comparison across federated learning types for multi-disease prediction from MIMIC-IV.

Horizontal and Vertical FL served as strong baselines, with Horizontal FL performing slightly better where data schemas were consistent across institutions. Although Split-FL showed slightly lower ROC-AUC scores, it preserved patient privacy more effectively by limiting data exposure through model-splitting. The overall results emphasize the trade-off between predictive performance and privacy: Multi-task and Personalized FL offer higher accuracy, while Split-FL ensures stronger data confidentiality with minimal performance compromise. These insights support the need for context-aware FL strategies in healthcare, where both accuracy and data protection are critical.

C. Cross-Hospital Generalization Across Evaluation Metrics

To assess the generalization capabilities of the CrypTen-FL framework, we performed a cross-hospital analysis across five conventional performance metrics—Area Under Receiver Operating Characteristic Curve (ROC-AUC), Accuracy, Precision, Recall, and F1-Score. For disease-specific prediction tasks, the model trained on a subset of hospitals, and then used observed data from an unseen institution, creating a scenario similar to realistic deployment situations where the clinical data are not identically distributed (non-IID). As summarized in Table II, CrypTen-FL achieved consistent cross-location performance, maintaining over 97.5% across all metrics when evaluated on hospitals not used in its training. A small decrease of .02977 (1.97%) was observed on ROC-AUC suggesting some decay in ranking ability; at the same time, Accuracy dropped by 2.09% and was sustained across locations. Values for Precision and Recall were similarly aligned, demonstrating balanced levels of specificity and sensitivity, and the F1-Score (which considers Precision and Recall) similarly showed a decrease of 2.11% confirming the model's robustness against changes in distributions.

Secure Multi-Party Computation (SMPC) protocols included in CrypTen were used for all assessments, guaranteeing that no patient-level information was disclosed during inter-institutional evaluation. In federated healthcare applications, where strict data protection laws must be followed, this privacy-preserving configuration is crucial. Because CrypTen-FL offers both dependable predictive performance and end-to-end privacy assurances, it has a significant potential for real-world adoption in decentralized clinical environments, as seen by the negligible performance deterioration noticed across unseen hospitals.

TABLE II. CRYPTEN-FL CROSS-HOSPITAL EVALUATION DEMONSTRATING STRONG GENERALIZATION CAPABILITY, WITH HIGH METRIC RETENTION AND MINIMAL PERFORMANCE DEGRADATION ON UNSEEN HOSPITAL DATA

Metric	Performance Retained (%)	Performance Drop (%)
ROC-AUC	98.03	1.97
Accuracy	97.91	2.09
Precision	97.50	2.50
Recall	97.51	2.51
F1-score	97.89	2.11

D. Comparison with Existing Models

With a training time of roughly 1.2 × the baseline and a computational overhead of only 8%, CrypTen achieves the best accuracy, roughly 94.2%, according to the comparative evaluation displayed in Table III. These findings show that even with the additional secure multiparty compute operations, CrypTen maintains model performance with no loss of efficiency. PySyft, despite achieving an accuracy of 92.8%, had a 15% overhead and almost 1.5 × the training time of the baseline, exhibiting the balance of computational complexity

against a more generalized privacy-preserving framework. TensorFlow Federated noted good performance for federated learning settings but had limited support for cryptographic security, evaluating a 93.5% accuracy at a 10% overhead and 1.3 × the baseline training time. In sum, the results imply that CrypTen attains the best trade-off precision, speed of computing, and privacy protection, which ultimately makes it a suitable framework for sensitive industries such as healthcare and finance.

TABLE III. COMPARATIVE ANALYSIS OF CRYPTEN, PYSYFT, AND TENSORFLOW FEDERATED IN TERMS OF MODEL ACCURACY, COMPUTATION OVERHEAD, AND TRAINING EFFICIENCY

Framework	Accuracy (%)	Computation Overhead (%)	Training Time (relative)	Remarks
CrypTen	94.2	+8	1.2× baseline	High accuracy with minimal loss; efficient SMPC integration
PySyft	92.8	+15	1.5× baseline	Broader privacy features but higher complexity and slowdown
TensorFlow Federated	93.5	+10	1.3× baseline	Strong FL accuracy, limited SMPC support, TensorFlow-only

E. Discussion

Using the MIMIC-IV dataset, this study proposes a privacy-preserving federated learning (FL) architecture for disease prediction across hospitals. We show that in multi-institutional healthcare settings, it is feasible to attain both strong data privacy and good predictive performance by combining decentralized key generation, CrypTen-based Secure Multi-Party Computation (SMPC), and differential privacy.

Split Learning continuously produced the best results out of all the FL paradigms that were studied, thanks to its capacity to jointly learn representations without disclosing raw data. Given the enhanced privacy guarantees and legal compliance (e.g. HIPAA, GDPR), the minor overhead (10–30%) introduced by SMPC and DP-SGD is justified.

The framework's robustness and fairness in real-world varied clinical settings were confirmed by its good generalization across a variety of hospital datasets under MIMIC-IV and the lack of notable performance differences across age, gender, and ethnicity subgroups. Crucially, Shamir's Secret Sharing eliminated the need for centralized trust models by enabling decentralized key creation, reducing the possibility of single-point failure or compromise.

Case Study: Privacy Preservation in CrypTen-FL Using Communication Overhead.

We carried out a quantitative trade-off study across five federated learning (FL) strategies: Horizontal FL, Vertical FL, Personalized FL, Multi-task FL, and Split-FL in order to thoroughly evaluate the trade-off between privacy preservation and operational viability in the CrypTen-FL framework. Three factors were used to evaluate each strategy: computation overhead, communication overhead, and privacy level (enforced by Secure Multi-Party Computation and optional Differential Privacy). The Quant Mapping Reference provides standardized numeric values to interpret qualitative privacy levels (ranging from 0.4 for Low to 1.0 for Very High) and overhead categories (0–100% range). These qualitative attributes were then mapped to quantitative values using normalized scales (e.g. privacy: 0.6–1.0; overheads: 0%–100%) to facilitate consistent comparison. The results in Table IV show strategies like Vertical FL and Split-FL achieved the highest levels of privacy (1.0). They also incurred significant system costs, with communication and computation overheads exceeding 75% due to secure feature alignment and encrypted activation exchange, respectively.

In contrast, Horizontal FL and Personalized FL offered a more practical balance. Horizontal FL attained high privacy (0.8) with moderate communication overhead (50%) and minimal computational burden (20%), making it suitable for environments with consistent feature schemas. By permitting local model customisation, personalized FL maintained strong privacy (0.8) and moderate computation (50%), while reducing communication overhead to 25%. Due to task-specific learning, multi-task FL had significant computing costs and modest privacy (0.6), despite being effective in some collaborative environments. All things considered, this analysis draws attention to the inherent trade-offs in federated learning architecture for medical AI systems, allowing researchers to choose approaches that are in line with resource availability, deployment requirements, and data sensitivity—particularly important in datasets that are sensitive to privacy, such as MIMIC-IV.

TABLE IV. QUANTIFIED COMPARISON OF PRIVACY AND OVERHEAD TRADE-OFFS IN CRYPTEN-FL STRATEGIES

FL Strategy	Privacy Level (0–1)	Communication Overhead (0–100%)	Computation Overhead (0-100%)	Analysis
Horizontal FL	0.8 (High)	50 (Moderate)	20 (Low)	Model updates encrypted via SMPC; efficient for similar feature spaces.
Vertical FL	1.0 (Very High)	75 (High)	80 (High)	Encrypted feature alignment across clients increases communication cost.
Personalized FL	0.8 (High)	25 (Low)	50 (Moderate)	Local personalization reduces shared data, enhancing privacy.
Multi-task FL	0.6 (Moderate)	50 (Moderate)	75 (High)	Shared representation learning; higher task-specific computation load.
Split-FL	1.0 (Very High)	90 (Very High)	90 (Very High)	Securely exchanges activations; offers strongest privacy with high cost.

V. Conclusion and Future Work

Using the MIMIC-IV dataset, this paper proposes a safe and private federated learning system for disease prediction across hospitals. As a potent instrument for secure multiparty computation (SMPC), CrypTen provides a centralized platform for machine learning that protects privacy. It enables cooperative model training across dispersed and heterogeneous environments while maintaining the confidentiality of sensitive data through the integration of cryptographic protocols with tensor-based operations. CrypTen shows that privacy-preserving learning can be both feasible and scalable by minimizing security and accuracy trade-offs, in contrast to conventional methods.

Additionally, because of its natural compatibility with PyTorch, academics and practitioners can more easily embrace it in real-world applications like healthcare, finance, and cross-institutional cooperation, where stringent data sharing regulations are essential. According to comparative research, CrypTen improves data security, compliance, and trust while offering competitive performance.

Future studies will concentrate on supporting multimodal clinical data, refining SMPC protocols to lower computational cost, and enhancing system scalability across bigger hospital networks. The usefulness and efficacy of the suggested framework will be further reinforced by improvements in dynamic involvement, training that is fair, and real-world clinical deployment. These guidelines seek to promote the use of decentralized, moral, and privacy-preserving AI in cooperative healthcare settings.

References

[1] Garriga, R., Mas, J., Abraha, S., Nolan, J., Harrison, O., Tadros, G., & Matic, A. (2022). Machine learning model to predict mental health crises from electronic health records. Nature medicine, 28(6), 1240-1248.

[2] Xiao, C., Choi, E., & Sun, J. (2018). Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. Journal of the American Medical Informatics Association, 25(10), 1419-1428.

[3] Hobensack, M., Song, J., Scharp, D., Bowles, K. H., & Topaz, M. (2023). Machine learning applied to electronic health record data in home healthcare: a scoping review. International Journal of Medical Informatics, 170, 104978.

[4] Bani Issa, W., Al Akour, I., Ibrahim, A., Almarzouqi, A., Abbas, S., Hisham, F., & Griffiths, J. (2020). Privacy, confidentiality, security and patient safety concerns about electronic health records. International nursing review, 67(2), 218-230.

[5] Chenthara, S., Ahmed, K., Wang, H., & Whittaker, F. (2019). Security and privacy-preserving challenges of e-health solutions in cloud computing. IEEE access, 7, 74361-74382.

[6] Yigzaw, K. Y., Olabarriaga, S. D., Michalas, A., Marco-Ruiz, L., Hillen, C., Verginadis, Y., ... & Chomutare, T. (2022). Health data security and privacy: Challenges and solutions for the future. Roadmap to successful digital health ecosystems, 335-362.

[7] Deimazar, G., & Sheikhtaheri, A. (2023). Machine learning models to detect and predict patient safety events using electronic health records: a systematic review. International Journal of Medical Informatics, 180, 105246.

[8] Hao, R., Xiang, Y., Du, J., He, Q., Hu, J., & Xu, T. (2025). A Hybrid CNN-Transformer Model for Heart Disease Prediction Using Life History Data. arXiv preprint arXiv:2503.02124.

[9] Tiwari, P., Colborn, K. L., Smith, D. E., Xing, F., Ghosh, D., & Rosenberg, M. A. (2020). Assessment of a machine learning model applied to harmonized electronic health record data for the prediction of incident atrial fibrillation. JAMA network open, 3(1), e1919396-e1919396.

[10] Ellis, R. J., Wang, Z., Genes, N., & Ma'ayan, A. (2019). Predicting opioid dependence from electronic health records with machine learning. BioData mining, 12(1), 3.

[11] Hu, Z., Wang, Z., Jin, Y., & Hou, W. (2023). VGG-TSwinformer: Transformer-based deep learning model for early Alzheimer's disease prediction. Computer Methods and Programs in Biomedicine, 229, 107291.

[12] Li, Y. et al. (2022). Hi-BEHRT: hierarchical transformer-based model for accurate prediction of clinical events using multimodal longitudinal electronic health records. IEEE journal of biomedical and health informatics, 27(2), 1106-1117.

[13] Amirahmadi, A., Ohlsson, M., & Etminani, K. (2023). Deep learning prediction models based on EHR trajectories: A systematic review. Journal of biomedical informatics, 144, 104430.

[14] Makarov, N., & Lipkovich, M. (2025). A transformer-based model for next disease prediction using electronic health records. The European Physical Journal Special Topics, 1-10.

[15] Nguyen, H. et al. (2023). Clinically-inspired multi-agent transformers for disease trajectory forecasting from multimodal data. IEEE transactions on medical imaging, 43(1), 529-541.

[16] Desai, R. et al. (2020). Comparison of machine learning methods with traditional models for use of administrative claims with electronic medical records to predict heart failure outcomes. JAMA network open, 3(1), e1918962-e1918962.

[17] Rahman, et al. Enhancing heart disease prediction using a self-attention-based transformer model. Scientific Reports, 14(1), 514.

[18] Li, H., et al. (2025). A multistage, multitask transformer-based framework for multi-disease diagnosis and prediction using personal proteomes. medRxiv, 2025-02.

[19] Singh, M. K., Singh, A. K., Choudhary, P., Singh, P., & Singh, A. K. (2025). A Smart System for Tracking and Analyzing Human Hand Movements using MediaPipe Technology and TensorFlow. In *Demystifying Emerging Trends in Green Technology* (pp. 201-218). Bentham Science Publishers.

[20] Abdulhameed, I. S., Al-Mejibli, I., & Naif, J. R. (2021). The security and privacy of electronic health records in healthcare systems: A systematic review. Turkish Journal of Computer and Mathematics Education, 12(10), 1979-1992.

[21] Singh, A. K., Singh, M. K., Chaoudhary, P., & Singh, P. (2023). Future technology: Internet of Things (IoT) in smart society 5.0. In *Intelligent Techniques for Cyber-Physical Systems* (pp. 245-265). CRC Press.

[22] Kawamoto, K., Finkelstein, J., & Del Fiol, G. (2023, March). Implementing machine learning in the electronic health record: checklist of essential considerations. In Mayo Clinic Proceedings (Vol. 98, No. 3, pp. 366-369). Elsevier.

[23] Ramakrishnaiah, Y., Macesic, N., Webb, G. I., Peleg, A. Y., & Tyagi, S. (2025). EHR-ML: A data-driven framework for designing machine learning applications with electronic health records. International Journal of Medical Informatics, 196, 105816.

[24] Basil, N. N., Ambe, S., Ekhator, C., Fonkem, E., & Nduma, B. N. (2022). Health records database and inherent security concerns: A review of the literature. Cureus, 14(10).

[25] Poongodi, T., Sumathi, D., Suresh, P., & Balusamy, B. (2020). Deep learning techniques for electronic health record (EHR) analysis. In Bio-inspired Neurocomputing (pp. 73-103). Singapore: Springer Singapore.

[26] Mishra, V., & Mishra, M. (2022). Privacy and security concerns with electronic health records-shreds of evidence from India. IMI Konnect, 11(3), 41-54.

[27] Gupta, M., Phan, T. L. T., Bunnell, H. T., & Beheshti, R. (2022). Obesity Prediction with EHR Data: A deep learning approach with interpretable elements. ACM Transactions on Computing for Healthcare (HEALTH), 3(3), 1-19.

[28] Marcus, J. L., Hurley, L. B., Krakower, D. S., Alexeeff, S., Silverberg, M. J., & Volk, J. E. (2019). Use of electronic health record data and machine learning to identify candidates for HIV pre-exposure prophylaxis: a modelling study. The lancet HIV, 6(10), e688-e695.

[29] Dang, Trung Kien, et al. "Federated learning for electronic health records." ACM Transactions on Intelligent Systems and Technology (TIST) 13.5 (2022): 1-17.

[30] Thakur, Anshul, et al. "Knowledge abstraction and filtering based federated learning over heterogeneous data views in healthcare." npj Digital Medicine 7.1 (2024): 283.

[31] Salim, Mikail Mohammed, and Jong Hyuk Park. "Federated learning-based secure electronic health record sharing scheme in medical informatics." IEEE Journal of Biomedical and Health Informatics 27.2 (2022): 617-624.

[32] Bebortta, Sujit, et al. "Fedehr: A federated learning approach towards the prediction of heart diseases in iot-based electronic health records." Diagnostics 13.20 (2023): 3166.

[33] Meduri, Karthik, et al. "Leveraging federated learning for privacy-preserving analysis of multi-institutional electronic health records in rare disease research." Journal of Economy and Technology 3 (2025): 177-189.

[34] CU, Om Kumar, and Sudhakaran Gajendran. "EHR privacy preservation using federated learning with DQRE-Scnet for healthcare application domains." Knowledge-Based Systems 275 (2023): 110638.

[35] Khaled, A et al. (2025). Leveraging MIMIC Datasets for Better Digital Health: A Review on Open Problems, Progress Highlights, and Future Promises. arXiv preprint arXiv:2506.12808.

[36] Qian, J. et al. (2025). Association between emergency department to intensive care units time and in-hospital mortality: an analysis of the MIMIC-IV database. BMJ open, 15(4), e090011

[37] van Damme, P., Löbe, M., Benis, N., de Keizer, N. F., & Cornet, R. (2024). Assessing the use of HL7 FHIR for implementing the FAIR guiding principles: a case study of the MIMIC-IV Emergency Department module. JAMIA open, 7(1), ooae002.

[38] Wang, H., Wu, S., Pan, D., Ning, Y., Li, Y., Feng, C., & Gu, Y. (2024). Comparison of different intensive care scoring systems and Glasgow Aneurysm score for aortic aneurysm in predicting 28-day mortality: a retrospective cohort study from MIMIC-IV database. BMC Cardiovascular Disorders, 24(1), 513.

[39] Sadeghi, S., Hempel, L., Rodemund, N., & Kirsten, T. (2024). Salzburg Intensive Care database (SICdb): a detailed exploration and comparative analysis with MIMIC-IV. Scientific reports, 14(1), 11438.

[40] Singh, P., Khan, S., Singh, Y. V., & Singh, R. S. (2022). A Secure and Stable Humanoid Healthcare Information Processing and Supervisory Method with IoT-Based Sensor Network. *Journal of Sensors*, *2022*(1), 8568540.

[41] Horvath, A. N., Berchier, M., Nooralahzadeh, F., Allam, A., & Krauthammer, M. (2023). Exploratory analysis of federated learning methods with differential privacy on MIMIC-III. arXiv preprint arXiv:2302.04208.

[42] Johnson, A. E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S.,& Mark, R. G. (2023). MIMIC-IV, a freely accessible electronic health record dataset. Scientific data, 10(1), 1.

[43] Sun, Y., He, Z., Ren, J., & Wu, Y. (2023). Prediction model of in-hospital mortality in intensive care unit patients with cardiac arrest: a retrospective analysis of MIMIC-IV database based on machine learning. BMC anesthesiology, 23(1), 178

[44] Kaminaga, H., Awaysheh, F. M., Alawadi, S., & Kamm, L. (2023, December). Mpcfl: Towards multi-party computation for secure federated learning aggregation. In Proceedings of the IEEE/ACM 16th international conference on utility and cloud computing (pp. 1-10).

[45] Knott, B., Venkataraman, S., Hannun, A., Sengupta, S., Ibrahim, M., & van der Maaten, L. (2021). Crypten: Secure multi-party computation meets machine learning. Advances in Neural Information Processing Systems, 34, 4961-4973.

[46] Budrionis, A., Miara, M., Miara, P., Wilk, S., & Bellika, J. G. (2021). Benchmarking PySyft federated learning framework on MIMIC-III dataset. IEEE Access, 9, 116869-116878.

[47] Kanagavelu, R., Li, Z., Samsudin, J., Yang, Y., Yang, F., Goh, R. S. M., & Wang, S. (2020, May). Two-phase multi-party computation enabled privacy-preserving federated learning. In 2020 20th IEEE/ACM international symposium on cluster, cloud and internet computing (CCGRID) (pp. 410-419). IEEE.

[48] Zhu, H. (2020). On the relationship between (secure) multi-party computation and (secure) federated learning. arXiv preprint arXiv:2008.02609.

[49] Byrd, D., & Polychroniadou, A. (2020, October). Differentially private secure multi-party computation for federated learning in financial applications. In Proceedings of the first ACM international conference on AI in finance (pp. 1-9).

[50] Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., & Zhou, Y. (2019, November). A hybrid approach to privacy-preserving federated learning. In Proceedings of the 12th ACM workshop on artificial intelligence and security (pp. 1-11).

Is Metaverse Technology Ready to Welcome Online Banking Users?

A Systematic Literature Review

Yantu Ma[1], Dalbir Singh[2], Siok Yee Tan[3]*, Meng Chun Lam[4], Ab Ghani Nur Laili[5], Haodong Guan[6], Fengjin Lei[7], Ahmad Sufril Azlan Mohamed[8]

Center for Software Technology and Management-Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, Selangor, Malaysia[1, 2, 6, 7]

Center for Artificial Intelligence Technology-Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, Selangor, Malaysia[3, 4]

UKM-Graduate School of Business, Universiti Kebangsaan Malaysia, Selangor, Malaysia[5]

School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang, Malaysia[8]

***Abstract*—The advent of the Web 3.0 era has precipitated the gradual emergence of Metaverse technology as the frontier of next-generation interactive technology transformation in banking. Nevertheless, a considerable disparity exists between the user experience of Metaverse banking in virtual interactive environments and traditional online banks. Consequently, in the nascent stages of Metaverse banking development, examining how existing banking users adopt Metaverse technology and exploring related controversies from an information systems implementation perspective is imperative. The present study employs a systematic literature review methodology, screening 19 relevant articles published between 2020 and 2024 from two leading academic databases, Web of Science (WOS) and Scopus. The review identifies factors hindering current bank users' adoption of Metaverse banking, such as privacy concerns and lack of social norms, while elucidating motivational drivers in Metaverse contexts, including usability and perceived enjoyment. Moreover, from a banking application perspective, this study proposes implementation recommendations for the initial deployment of Metaverse banking, including lowering user adoption barriers and enhancing immersive experiences. The study draws upon contemporary customer needs to analyse key decision-making considerations when using Metaverse banking services. It highlights key adoption barriers and drivers for sustained usage, emphasising potential challenges in the initial development phase, such as privacy leaks and user behaviour monitoring in virtual environments. Furthermore, it identifies unexplored research gaps in the early implementation stage of Metaverse banking. This review synthesises current knowledge on Metaverse technology in banking and outlines practical considerations and strategic directions for its integration within the industry.**

***Keywords*—Banking; human-computer interaction; information systems; Metaverse; virtual reality**

I. Introduction

At present, Metaverse banking has sparked widespread academic discussion [1]. This study, employing a systematic literature review methodology, evaluates the primary adoption issues faced during the early technological implementation stage of Metaverse banking. Following a series of structured procedures inherent to systematic reviews, the study traces the origins of Metaverse banking by combining empirical research on online banking users with the discourse surrounding Metaverse technologies, thereby exploring both the challenges and advantages of Metaverse banking from an adoption perspective.

Metaverse banking represents a frontier business strategy integrating Metaverse technologies with online banking services [2]. By leveraging the advantages of the Metaverse, it aims to enable users to access banking services anytime and anywhere through a decentralised approach [3]. It uses immersive technologies to simulate the face-to-face interactions of traditional offline banking [4], thereby reducing the distance-related costs of accessing banking services [5]. Additionally, it seeks to enhance the connection between users and banks while preserving the high-quality service traditionally associated with physical bank branches[6].

In 2021, Facebook's rebranding to Meta by its CEO, Mark Zuckerberg, sparked intense debate within the retail industry regarding Metaverse banking [1]. Financial institutions such as JPMorgan Chase, HSBC, and IBK have already begun experimenting with this new paradigm by purchasing land and establishing virtual platforms within digital worlds [7]. The decentralised architecture and immersive experiences of the Metaverse are expected to empower banks to deliver more efficient business solutions, reduce labour costs, and strengthen customer relationships. However, research on Metaverse banking remains mainly exploratory, with a limited investigation into users' technology adoption. In the financial sector, user acceptance has been identified as a critical factor [4]. The unfamiliarity of technology induces concerns and apprehension among users [8], and the current lack of empirical research presents early risks and challenges for implementing Metaverse banking. For enterprises, this translates into significant uncertainty, which is currently viewed as a major barrier to the development of Metaverse banking.

From the initial review of the literature on Metaverse Banking, this research finds that the service quality of online banking, as the precursor of Metaverse Banking, has been extensively examined. The service quality of online banking

*Corresponding author

encompasses users' final adoption decisions and reflects their perspectives and perceptions under a given technology. Within the scope of this study, such findings provide potential functional insights into linking Metaverse technologies with the banking sector. Accordingly, this study employs service quality as the primary retrieval criterion to explore research on Metaverse Banking in its early stage of technological development, thereby analysing users' perceptions of technology adoption from the perspective of Metaverse Banking's initial evolution.

II. Motivation

Metaverse technology has been recognised as a frontier of the Web 3.0 revolution [1], widely anticipated by both the market and academia. It is expected to empower the development of multiple technologies [5], providing industries with more efficient solutions and delivering higher-quality technological services to the public. However, studies have indicated that 85% of users have not engaged with the Metaverse or related virtual technology products [4].

Despite the transformative potential of Metaverse technology in redefining the connection between users and business operations, research on financial technology adoption emphasises that such innovations should enable users to integrate with financial systems [9] seamlessly. A review of the current state of research on Metaverse banking reveals a gap in studies addressing its early-stage implementation [7]. Reviewing the current research on Metaverse banking reveals several perspectives[10]. Discuss the impact of Metaverse technologies on various industries and their future opportunities and prospects[11]. Further examination, from a business perspective, of the potential applications and challenges of the Metaverse in commerce and trade will be conducted[12]. Explore the possibilities of decentralisation, future directions, opportunities and challenges empowered by Metaverse technologies[4]. Investigate the potential and challenges of the banking sector development[5]. Propose possible directions and conceptual frameworks for future research. Thus, the study finds that existing research has yet to sufficiently address the implementation of Metaverse banking in its early stages, as most literature primarily focuses on discussions of the technology's future trajectory. Such a gap introduces significant uncertainty regarding its initial integration into banking operations [10]. This highlights the urgent need for academia to devote greater attention to and engage in more profound exploration of Metaverse banking.

Clarifying the early implementation challenges of Metaverse banking will accelerate its integration into financial services and advance the banking sector's transition into the Web 3.0 era.

III. Methodology

The systematic literature review (SLR) method employed in this study is illustrated in Fig. 1. By clearly defining motivation, research questions (RQs), and keywords, the study further identified the electronic databases used for retrieval. This research draws on the three-stage approach described by Kitchenham and Charters, combined with the SLR practices from Brereton's experience in software engineering [13].



Fig. 1. SLR method.

A. Research Questions

As the banking industry continues to accelerate the exploration and implementation of virtual financial services, Metaverse banking technologies are gradually becoming a focal point for transforming financial service models. However, existing research provides limited clarity on the key factors influencing banking users' adoption decisions regarding Metaverse technologies. This lack of clarity introduces uncertainty into the implementation process for financial institutions, potentially leading to deviations in resource allocation and strategic decision-making. Without identifying the core factors driving customer adoption, implementing Metaverse banking will likely incur higher resource costs and decision-making risks. Based on these justifications, below are the specific research questions:

1) RQ1: What are the key factors determining user adoption of Metaverse Bank?

Identifying the decisive factors influencing users' adoption of Metaverse banking will enhance the precision of Metaverse technology implementation within the banking sector. It will provide strategic guidance for banks engaging in Metaverse initiatives, help mitigate uncertainties during the early stages of development, and offer user-centred recommendations to support effective implementation from the perspective of banking customers.

The emergence of Metaverse technology has driven a transformation in the service model of the financial industry. Within the current research context of Metaverse banking, the Metaverse presents opportunities and significant challenges to the banking sector. While its advanced interactive capabilities and immersive user experiences offer novel possibilities, they also introduce substantial differences from traditional online banking services, particularly in the early stages of Metaverse banking implementation. These differences may heighten users' sense of uncertainty and concern. Consequently, in the initial phase of Metaverse banking adoption, users' perceptions of banking services become a critical focus of analysis. Therefore, these three research questions arise on the adoption of the Metaverse for banking services:

2) RQ2: What is the current adoption status of Metaverse banking in its initial development stage?

Metaverse technology has brought transformative user experiences and business models to the banking industry,

driving financial services toward immersive and interactive application trends [14]. Although some banking institutions have initiated exploratory attempts, research on user adoption of Metaverse banking in its early development stage remains scarce. Metaverse banking presents significant differences in user experience and service delivery compared to traditional banking technologies, potentially making it difficult for long-term users of conventional technologies to learn and adapt [15] quickly. Therefore, to promote the empowerment of the banking industry through Metaverse technologies, this study aims to explore the user adoption status of Metaverse banking during its initial development phase.

The development of technology often participates in the continuous empowerment of fixed markets. In the long-term evolution of the banking industry, the fixed needs of its users have already been well established. As Metaverse technology gradually integrates into the banking sector, user adoption perspectives toward traditional banking service technologies have become increasingly critical. The views of user groups who were the audience of previous technological applications facilitate the rapid integration of new technologies into the industry and enhance the success rate of these technologies in empowering business functions. At the same time, by maximising the mitigation of user concerns arising in the early stages of new technology development, such perspectives also shape the initial impression of new technologies within the user's field of perception.

3) RQ3: What are the implementation suggestions for bank users to promote the adoption of the Metaverse in its initial development stage?

The current research lacks guiding recommendations regarding how Metaverse technologies can effectively empower banking services during the initial implementation stage [7]. At the same time, systematic analyses focusing on users' financial needs are scarce, and research on the implementation pathways of Metaverse banking remains in an early exploratory phase, leaving a theoretical gap for practical reference in the banking industry's early-stage development [1]. Therefore, this study will take traditional banking users' adoption cases as the primary reference to explore user feedback on adopting Metaverse technologies in banking services and provide targeted and actionable guidance for the initial implementation of Metaverse banking.

In empowering banking services through the Metaverse, Metaverse banking remains characterised by immature technology, a lack of standardised protocols, and limited user awareness in early technological development. From the perspective of financial technology adoption, the successful adoption of new technologies should ensure a seamless user experience while meeting the performance demands of financial services. Therefore, the challenges encountered during the early development phase may offer valuable guidance for the initial implementation of emerging technologies, thereby reducing trial-and-error costs and implementation barriers for banks.

4) RQ4: What are the potential challenges for banks in promoting Metaverse banking during their initial development stage?

As the next generation of financial technology transformation, Metaverse banking faces numerous challenges at the initial stages of industry and technological development, characteristic of an early-cycle emergence. Most existing studies focus on conceptual discussions and potential forecasts of Metaverse banking, while research analysing the actual challenges and barriers encountered by banking users during its early development remains scarce. User acceptance is a critical milestone for the technological maturity of Metaverse banking [4]. Therefore, this study explores the challenges users face in adopting Metaverse banking during the initial development stage, filling the theoretical gap and providing strategic support for integrating Metaverse technology into the banking industry.

B. Keywords Search

Based on the RQs, this study requires a review of the characteristics of online banking technologies and Metaverse technologies and their corresponding user needs. Therefore, the keywords "Metaverse" and "Online Banking" were initially selected. Given that both Metaverse and online banking fall under online services, this study adopts the optimised online service quality framework proposed by [16], which includes explicit dimensions relevant to the banking industry. This ensures the methodological relevance of online banking within this study. Accordingly, the dimensions of responsiveness, efficiency, fulfilment, reliability, security, and website attributes were included [17].

However, due to potential ambiguity in the retrieval of the term "website attributes" and based on interpretations in previous studies, this research replaces it with the terms "usability" and "ease of use." As a result, the final set of retrieval keywords is defined as:

("Metaverse" OR "online banking") AND ("security" OR "availability" OR "usability" OR "privacy" OR "efficiency" OR "responsiveness" OR "fulfilment" OR "reliability").

C. Database Search

The databases used for this systematic literature review include Web of Science (WoS) and Scopus. From an authoritative perspective, WoS is one of the most widely used databases in academic research, while Scopus offers a broad collection of mainstream literature. Using these two databases, the initial search yielded 2,184 articles. The preliminary screening criteria included publications from the past five years, English language, and document types limited to articles and review papers. The resulting sample size is considered sufficient for analysis.

D. Inclusion Criteria and Screening Process

The initial search formula and databases were used to retrieve 5,823 documents. After the implementation of screening criteria predicated on publication year (2020–2024) and document type (journal articles and review papers), the number was reduced to 2,435. To ensure the uniformity of the research sample, additional restrictions were implemented to retain exclusively English-language publications, which resulted in the final selection of 2,184 research articles. Subsequently, Rayyan automated screening tools were employed to remove duplicates, retaining 1,375 preliminary research samples. The sample size for this systematic literature review is illustrated in Fig. 2.

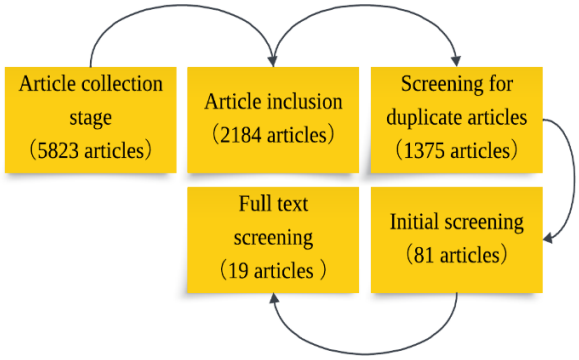


Fig. 2. Sample selection process for this systematic literature review.

In Fig. 2, during the title and abstract screening phase, the research prioritised including research in which titles and abstracts potentially referenced the Metaverse technology's effectiveness and technical characteristics. Concurrently, articles that primarily focused on underlying technical principles and development were excluded from the analysis. Empirical studies examining online banking user needs were included, while research centred on single functionalities or business impacts was excluded. Eighty-one documents were identified as being directly pertinent to the research objectives. During the full-text screening phase, a review of the relevant literature revealed that many studies primarily focused on exploratory research of Metaverse technology in third-party applications (e.g. education; gaming) or retrospective studies on online banking. Consequently, the present study exclusively retained literature analysing the adoption of online banking technology and included review studies evaluating the effectiveness of Metaverse technology. After thoroughly examining the full texts, 19 articles were selected for inclusion in the final analysis.

IV. Result

Thirteen studies indicate a negative relationship between users and perceived risk in online banking. Perceived risk factors have been identified as significant barriers to adopting banking technologies. Among these studies, two were sourced from the Web of Science (WoS) database and eleven from the Scopus database. Notably, [18] provides a more detailed explanation of the dimensions of perceived risk.

A total of eight studies explain the mediating factors and relationships between adoption intention and usage attitude, including two from the Web of Science (WoS) database and six from the Scopus database.

In the context of the Metaverse, five studies address the barriers Metaverse users face, explaining the indirect factors that hinder user adoption. Among these, four studies were sourced from the Scopus database and one from the Web of Science (WoS) database.

A total of four studies identify the usage motivations of Metaverse users, highlighting the indirect factors that promote these motivations. Among them, three studies were sourced from the Scopus database and one from the Web of Science (WoS) database.

TABLE I. Results Organization

Described content		
Section 4.1	Subsections 4.1.1	Perceived risk of bank users
	Subsections 4.1.2	Banking users' adoption intentions
	Subsections 4.1.3	Bank users' attitudes
Section 4.2	Subsections 4.2.1	Factors that hinder banking users' adoption intentions
	Subsections 4.2.2	Factors that mainly drive bank users' adoption decisions
	Subsections 4.2.3	Factors driving user adoption decisions
Section 4.3		Metaverse Bank Current User Adoption Factors

The organisation of this study's chapters is outlined in Table I. This study's literature review indicates that users' intention primarily drives their decisions regarding adopting Metaverse banking. Section 4.1 of this study identifies the critical factors influencing users' formation of adoption intentions, categorising these factors into positive and negative sub-factors. Section 4.2.1 discusses the sub-factors contributing to negative perceptions; within these negative factors, negatively correlated factors represent the key determinants influencing users' consideration of adopting banking services, whereas other negative factors impact users' motivation to adopt. Conversely, Section 4.2.2 examines positively correlated factors, which primarily facilitate users' adoption intentions, unlike negative factors. Additionally, Section 4.2.3 discusses in greater detail both the positively correlated factors and their impacts on users' adoption intentions.

A. The Formation of the Adoption Motivation of Banking users

In online banking, three key components primarily influence user adoption factors. These conclusions are drawn from multiple empirical studies rather than theoretical constructs or definitions. Literature categorises these components as follows:

1) Perceived risk: Perceived risk is identified as a critical barrier to the adoption of banking technologies [18]. In 15 reviewed studies, it is consistently reported that perceived risk negatively influences online banking user adoption. Users' perceptions of risk lead to harmful adoption intentions and attitudes, which hinder their actual adoption behaviour in the banking sector [19].

2) Adoption intention: Adoption intention refers to an individual's subjective willingness to use a new technology or service in the future. It reflects the extent to which a person is prepared to take actual usage actions. Across the 15 studies reviewed, the adoption of banking technologies was found to be primarily driven by users' intentions. The examined studies largely focus on contexts in India, Tunisia, and various parts of Europe, highlighting that users' intention to adopt is the primary driver of whether they choose to use banking services [20].

3) Usage attitude: Usage attitude refers to an individual's positive or negative disposition toward a technology or service during usage. In the 15 studies investigating factors that

promote the adoption of online banking, usage attitude is consistently found to have a positive influence on adoption intention. This attitude is often shaped by previous usage experiences or perceptions formed in non-contact scenarios with the technology [21]. Usage attitude determines whether users maintain a positive perspective toward the technology before use.

Given the specific nature of the financial industry, users' adoption decisions are mainly driven by concerns over security and risk. Whether the technology is ultimately used depends on adoption intention, while the user experience is primarily shaped by usage attitude.

B. The Composition of Factors that Motivate the Adoption of Metaverse Banking

The publications in this review describe various factors promoting the adoption of Metaverse banking. According to the findings of this study, several factors positively influence users' usage attitudes, and many of these factors simultaneously enhance users' adoption intentions through their impact on usage attitudes. Perceived risk emerges as the primary and only negatively correlated factor hindering banking sector customer adoption [18]. It is reviewed in the following sections.

1) The negative factors adopted by the Metaverse banking industry: According to the study by [18], the perceived risk of banking users comprises six dimensions: privacy, security, social, financial, time, and performance risks. These dimensions collectively form a comprehensive assessment of potential technological risks made by users before adopting online banking technologies.

Privacy risk primarily refers to the potential for personal information leakage during online banking services. This type of risk plays a critical role in determining whether users will continue to use banking services [22], [23]. Security risk involves external infringements that may threaten banking services, and this risk typically causes substantial disruption to users' perceptions. Social risk manifests as the potential relational risks arising from users' interactions and communications with banking staff during online banking transactions [19], [21], [24].

The performance risk reflects user concerns that the bank's services or products may not meet expected performance standards. This is particularly relevant in online environments, where system failures or technological weaknesses may lead to adverse outcomes [25]. Financial risk encompasses the initial economic expenditures in purchasing banking services and potential additional financial burdens incurred during service maintenance and continued usage. Time risk refers to the excessive time required for online banking services [20]. This includes time spent navigating the banking platform, learning to operate system functions, and correcting transactional errors.

These dimensions of risk collectively influence users' overall perception of risk in online banking and subsequently affect their technology adoption decisions [18], [26], [27].

The negative factors influencing users' adoption of Metaverse banking are presented in the Fig. 3. A Metaverse design lacking social functionality makes it difficult for users to receive feedback and interaction support from others, a critical component in replicating real-world experiences [28], [29]. The absence of social features weakens users' sense of social presence within the Metaverse environment, negatively affecting their motivation to use the system[30]. Furthermore, if implementing Metaverse technology leads users to perceive it as overly complex, it may hinder their ability to manage banking-related tasks effectively. When users encounter vulnerabilities in banking services [29], this perceived complexity can suppress their willingness to accept such services and intensify psychological barriers related to uncertainty and operational difficulty.

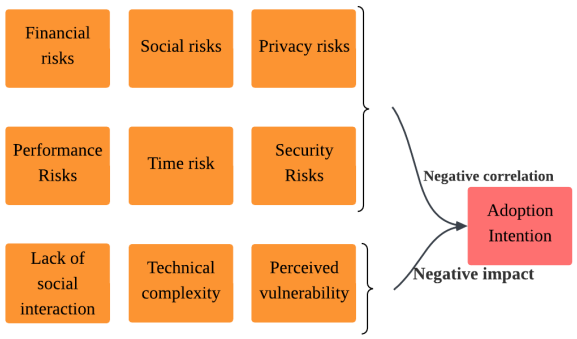


Fig. 3. Negative factors affecting the current user adoption of Metaverse banking.

2) Positive correlation factors for the adoption of Metaverse banking users: Perceived usefulness and perceived ease of use in online banking have been identified as positively correlated factors influencing customer adoption in 13 studies [28]. Perceived usefulness is the user's subjective assessment of how well the current technology can help solve practical problems. Perceived ease of use refers to the degree to which banking users subjectively believe that using a banking technology platform requires minimal effort. Usefulness and ease of use determine users' adoption intentions and are positively associated with their usage attitudes [31].

Perceived usefulness and perceived ease of use are two key determinants of whether users adopt banking technologies, and they comprise several sub-factors [27], [28]. Perceived usefulness is mainly driven by perceived ease of use and privacy [25]. Perceived ease of use refers to the extent to which users perceive the system as easy to use and whether they can conduct business in a way they consider simple [18], [21]. Privacy reflects the degree to which users believe their personal information is secure.

Perceived ease of use is also shaped by government support and system availability [21], [28]. Government support manifests in users' perceptions that banking platforms endorsed or supported by government authorities are easier to use [19], [28], [32]. Availability refers to how quickly users can learn and operate the system effectively, with minimal errors, thereby enhancing their overall user experience [33].

The positively correlated factors influencing user adoption are presented in Fig. 4. A positive correlation exists between the

immersive experience of Metaverse technology and users' intention to use it [29], similar to the relationship observed with perceived usefulness and ease of use. Additionally, users' sense of social presence within the Metaverse is positively related to their immersive experience [29]. This implies that a strong sense of social presence in the Metaverse environment enhances users' immersion, which fosters their intention to adopt Metaverse technologies.

Furthermore, system interoperability is positively associated with social presence [20], [24]. Interoperability enables users to interact more easily and intuitively with the platforms provided by Metaverse, thereby increasing their overall acceptance of Metaverse banking services. User perceptions of curiosity, enjoyment, and self-efficacy also enhance perceived ease of use regarding Metaverse banking technologies [27]. At the same time, perceived enjoyment and social norms positively influence perceived usefulness. The synergistic effect of these multiple factors significantly improves users' positive attitudes toward online banking services and promotes the adoption of new technologies from various perspectives [25], [26].

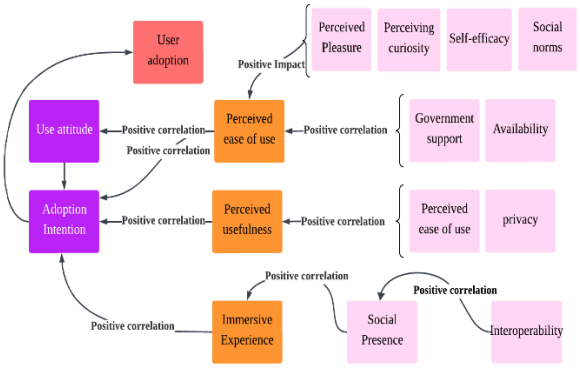


Fig. 4. Positively correlated factors affecting current user adoption of Metaverse banking.

3) Positive factors for the adoption of Metaverse banking users: Performance and effort expectancy influence users' adoption of online banking services [20]. Performance expectancy is the degree to which users believe using a particular service or system will effectively help them achieve their goals [20], [34]. This perception primarily stems from users' assessments of the service's practical utility. Existing studies have pointed out the notable advantages of online banking in terms of convenience and cost-effectiveness—for example, enabling users to manage their accounts efficiently anytime and anywhere at lower costs—which fosters positive performance expectations and, in turn, increases users' willingness to adopt financial technologies such as mobile payments and online banking.

Meanwhile, effort expectancy refers to users' subjective judgment of the ease of use of a new technology or service before adoption. Banking technologies encompass system simplicity, complexity, and comprehensibility. Research consistently shows that ease of use can effectively stimulate users' positive attitudes toward technology and significantly enhance their intention to use it. For instance, studies by Gupta and Arora on mobile payment adoption among Indian users [19], [33] on Internet banking in Jordan, and [35] on mobile banking services in Iran's private banks all confirm that users are more likely to adopt banking services when they perceive them as easier to use.

In summary, performance and effort expectancy jointly drive users' decisions to adopt online banking services [20]. In online banking, facilitating users' adoption intentions includes lowering usage costs, which reduces economic barriers and brings users closer to banking services by lowering the cost of usage. Simple usage enhances users' self-efficacy, helping them believe they can operate banking services independently, reducing unfamiliarity and fear of the technology and strengthening their motivation to use it. Banks should establish a positive brand image to enhance credibility and market recognition, build strong social influence to provide external support for user decision-making and enhance perceived enjoyment during service interactions to elicit positive emotional responses. Proactively identifying and responding to user needs encourages product innovation driven by users, increasing satisfaction and accumulating positive experiences, which reinforce users' continued usage intentions. Ultimately, this strengthens perceived trust and creates a positive feedback loop for the bank's adoption strategy [24], [29].

The positive factors are illustrated in Fig. 5. The Metaverse, with its unique interface and interactive mechanisms, also positively influences users' adoption intentions [36]. The degree of anthropomorphism in Metaverse avatars facilitates emotional connections during user–system interactions [37], leading banking users to perceive the system as a service entity with human-like characteristics [34]. This perception reduces users' alienation toward technology and stimulates their willingness to explore it actively.

The immersive experience the Metaverse offers enhances users' alignment with platform functionalities and usage scenarios by simulating realistic environments and interactive contexts. This deeper engagement actively guides users to participate in banking processes [36]. Furthermore, the realism conveyed by the Metaverse increases users' perceived credibility of Metaverse banking services, reinforcing the rationality and goal relevance of their usage behaviour and strengthening their positive expectations toward adoption [38].

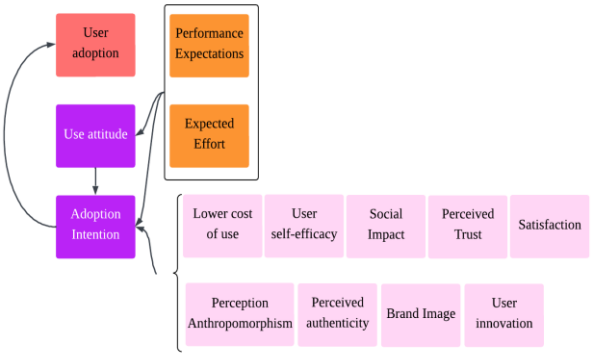


Fig. 5. Positive factors influencing Metaverse banking's current user adoption.

C. Summary of user Adoption Factors for Metaverse Banking

The factors influencing Metaverse user adoption are presented in Table II. In the early decision-making process of user adoption in the Metaverse, perceived risk is a set of negative factors that initially trigger user concerns. Perceived risk exerts a persistent and adverse effect on banking users' adoption intention and actual behavior [20]. When facing new financial technologies, users tend to evaluate risks, and uncertain perceptions of privacy breaches, account security, economic stability, or information reliability may suppress adoption motivation [24]. This undermines users' trust in virtual technologies and reduces their perceived controllability over operations, intensifying their tendency to avoid engagement.

The negative factors in Metaverse technology tend to shape a fixed usage impression during users' initial contact with Metaverse banking. This results in increased operational complexity, frustrated initial experiences, and decreased service satisfaction. Such unfavourable first-time experiences weaken users' perception of the platform's functional value, reducing their intention to continue using the service [25]. Unlike perceived risk, these factors do not directly cause users to reject Metaverse banking but instead suppress the drivers of long-term usage.

Banking users' subjective perception of usability determines the psychological threshold for their continued use of Metaverse banking. When the operational paths are intuitive, the interface logic is clear, and the business process feedback is smooth, users' cognitive load is reduced to an acceptable level, enhancing their sense of self-efficacy and laying a positive foundation for usage [28]. On this basis, users' evaluation of the usefulness of Metaverse banking is reflected in their overall judgment of efficiency and outcomes in achieving financial objectives. When virtual banking demonstrates precise alignment with real-world needs in fund management, risk alerts, and value-added services, users are more likely to regard it as a necessary and trustworthy financial service. The highly realistic spatial presentation, contextualised data visualisation, and real-time interaction in the Metaverse environment encourage users to focus on immersive experiences, enhancing their usage satisfaction and fostering emotional bonds that drive their intention for continued use [36]. This, in turn, promotes establishing a long-term and sustainable relationship between users and the bank.

TABLE II. METAVERSE BANKING ADOPTION FACTORS

Factor support	Factor name	Factor definition
Perceived risks in online banking services	Privacy risk	Users' concerns about the leakage of their personal information
	Safety risk	Users are worried about their losses due to system vulnerabilities and cyberattacks.
	Social risk	Users are concerned about the negative factors from outside society when using banking services.
	Financial risk	Users' concerns about the impact of banking services on their property safety
	Time risk	Users' concerns about time loss caused by banking factors during the process of using banking services
	Performance risk	Users' concerns about the impact on business progress due to system performance
Negative factors in the Metaverse	Metaverse social interaction	The Metaverse banking needs to have a social interaction function with others.
	The perceived complexity of the Metaverse	Users' judgment of the complexity of a Metaverse bank before they come into contact with it
	The perception of the Metaverse is fragile	The Metaverse banking needs to ensure the stability of its services
Perceived usability	Government support	Support from authoritative institutions
	Availability	The degree to which users can enjoy quick, stable and straightforward business services
	Perceived pleasure	The extent to which users enjoy the process of handling business
	Perceive curiosity	The degree of users' curiosity about the use of banking services
	Self-efficacy	Adopt simple designs to drive the user's design flow.
	Social norms	Legal regulations of the Metaverse society
Perceived usefulness	Perceived usability	User-friendly design that enables users to experience quickly
	Privacy	Formulate privacy protection policies for users.
Immersive experience	Social existence	The user's recognition of their virtual self in the Metaverse
	Interoperability	The ease of use for users to operate virtual avatars in the Metaverse
Adoption intention (positive impact)	Expected effort	The user's initial expectation of effort before use
	Expected performance	The user's initial expectation of performance feedback before use
	Low-cost value	Lower-cost use value
	User self-efficacy	The user believes that they can independently complete the business.
	Social influence	Users are externally influenced by their attitudes and behaviours in external social relationships.
	Perceived trust	The sense of trust that users have towards banks in the absence of information
	Satisfaction	The user's satisfaction with the actual use
	Perceptual personification	Feel the degree to which the avatars in the Metaverse banking are similar to real humans.
	Perceived authenticity	The degree to which Metaverse banking is similar to those in the real world
	Brand image	Users' impression of the brand
	User innovation	Use the degree to which user demands are met.

V. RESEARCH QUESTION DISCUSSION

A. RQ1

The key factors influencing Metaverse user adoption are presented in Table III. Perceived risk among traditional banking customers refers to the user's subjective anticipation of potential losses or adverse consequences from using specific technologies or banking services. Perceived risk poses a direct barrier to adopting Metaverse banking technologies. The higher the perceived risk, the lower the likelihood that users will adopt such technologies, significantly suppressing their motivation to engage [31]. In contrast, adoption intention represents the positively correlated factors that drive users toward usage. These factors are primarily shaped by users' previous experiences with technology and their psychological evaluations of current systems. Usage attitude, in turn, functions as a reinforcing

mechanism that enhances users' positive perceptions of banking services and mitigates negative resistance [27], thereby indirectly facilitating the transformation of intention into actual adoption behaviour.

TABLE III. KEY FACTORS IN USER ADOPTION OF METAVERSE BANKING

Factor effect	Factor Description
Perceived Risk	Negative factors that hinder user adoption
Adoption Intention	Positive factors driving user adoption
Attitude	Positive factors that promote user adoption

From the perspective of Metaverse banking, perceived risk is a critical barrier in users' adoption decisions, influencing whether users will further consider engaging with banking services [29]. Adoption intention is the central determinant of user behaviour, directly guiding whether users proceed to adoption. In contrast, usage attitude plays a supportive role by positively shaping users' evaluations and acceptance of the technology, thereby indirectly influencing adoption intention and the final decision to adopt

B. RQ2

In the current state of Metaverse user adoption, as presented in Table IV. Privacy risk and perceived privacy reflect users' concerns regarding privacy dimensions in the context of Metaverse banking and their expectations for privacy protection [18], [19].

Security risk and financial risk represent users' concerns about the safety of banking systems and the potential for financial loss. Specifically, users have expressed apprehension about the economic risks of Metaverse banking technologies [18]. The more unrestricted usage of Metaverse technology may further deepen users' concerns about such risks.

Social risk, influence, brand image, and government support represent external environmental factors impacting user adoption. Social risk emphasises users' fear of encountering adverse social outcomes, while social influence and brand image relate to the impact of social norms and platform reputation on adoption decisions. Government support refers to users' recognition of a banking platform's legitimacy, even without prior usage experience, due to endorsement by authoritative institutions. These factors influence users' adoption decisions, highlighting the role of external social evaluations in shaping users' willingness to adopt [18].

Performance risk and performance expectancy capture users' concerns and aspirations regarding the stability and effectiveness of system performance. Banking users expect reliable platforms that consistently support business operations. Frequent technical issues contribute to users' sense of vulnerability and erode trust in Metaverse banking, thereby hindering adoption behavior [25].

Social interaction, social presence, and interoperability are interdependent in the early stages of Metaverse development. Together, they form the basis of user interaction behaviours and subjective system perceptions in Metaverse banking [29]. A well-designed social system within the Metaverse contributes to users' sense of belonging, trust, and immersive experience in a virtual social environment.

From the perspective of Metaverse banking, perceived complexity, perceived ease of use, usability, and effort expectancy, though originating from different research perspectives, all fundamentally refer to the perceived ease of use and learnability of a technological system. These factors reflect users' psychological burden and behavioural threshold during learning and usage. The immersive mode offers further possibilities for user education and interactive engagement[30]. Designing systems that align with users' expected effort levels and offering intuitively easy-to-use features can reduce system complexity and facilitate adoption decisions.

Perceived enjoyment and perceived curiosity relate to users' psychological satisfaction and anticipation of user experiences. Perceived curiosity primarily stimulates interest and the desire for novel experiences [29], while perceived enjoyment pertains to emotional fulfilment during use. Both contribute to users' adoption intentions and attitudes as feedback mechanisms that enhance user engagement in Metaverse environments [36].

Perceived anthropomorphism and perceived realism, while focusing on the human-like expression of the platform and the authenticity of the interaction environment, aim to assess the realistic experience the Metaverse offers [39]. Lifelike anthropomorphic designs and authentic user experiences promote the intention to adopt Metaverse technologies, enhancing immersive experiences and emotional trust in Metaverse banking services and strengthening adoption intentions and continued usage behaviour.

Satisfaction and user innovativeness stem from perspectives of experiential evaluation and individual tendencies toward adopting new technologies. Both reflect users' feedback on technology experiences and exploratory motivation [21]. Satisfaction represents the overall emotional evaluation of current usage outcomes and is a key precursor to continued use or recommendation behaviour. User innovativeness captures individuals' willingness to try and adopt emerging technologies. In the design of Metaverse banking systems, continuous user feedback should be used to optimise the system, thereby enhancing user satisfaction iteratively.

TABLE IV. THE STATE OF METAVERSE BANKING ADOPTION

Factor effect	Name of homogeneous factor
Negative correlation factor	Privacy risks and privacy
	Security risks and financial risks
	Social risk, Social influence, Brand image, Government support
	Time risk
	Performance risk and expected performance: The fragility and trust of perception in the Metaverse
Positive correlation factor	Perceivable ease of use, usability, expected effort, and user self-efficacy
Negative influencing factors	Metaverse social interaction, social presence, and interoperability
	The perceived complexity of the Metaverse
Positive influencing factors	Perceive pleasure and perceive curiosity.
	Low-cost usage threshold
	Social norms
	Perceptual personification and perceptual authenticity
	Satisfaction and user innovation

C. RQ3

The implementation guidelines for Metaverse banking are suggested in Table V. In the early development stage of Metaverse banking, customer willingness to adopt Metaverse technologies remains in a sensitive phase. At this stage, perceived enjoyment and curiosity can be effective facilitators between banks and users, smoothing the adoption process. In particular, perceived curiosity motivates users to explore and engage with Metaverse banking.

The immersive advantages of the Metaverse offer significant technical potential for implementation in banking services. Perceived realism and perceived anthropomorphism both contribute to users' willingness to adopt Metaverse technologies [29]. During the initial implementation phase, ensuring that the Metaverse banking experience closely resembles that of the real-world banking environment will further support the adoption decisions of traditional banking users [30]. By embedding the operational flows of conventional banking services, the system allows users to engage with familiar banking processes. This reduces the need for extensive user education on the new technology and fulfils users' expectations for banking services. It also enables users to conduct banking activities in a familiar setting, enhancing their sense of self-efficacy through the capabilities of Metaverse technology.

Metaverse banking should adopt low-cost usage models to lower the entry barriers for customer participation, enabling a broader user base to engage with the technology in its early stages [29]. More inclusive design approaches should also be implemented to accommodate users with diverse characteristics. Leveraging the scalability of Metaverse functionalities will allow diverse user groups to efficiently and seamlessly complete transactions in the Metaverse banking environment.

TABLE V. METAVERSE BANKING ADOPTION RECOMMENDATIONS

Positive factor	Initial implementation suggestions
Perceived usability	Adopt a usage method that users are familiar with or can quickly understand
Expected effort	Adopt business processes that meet users' expectations.
Self-efficacy	Implement business processes that are sufficient to make users confident.
Availability	Adopt a more compatible design.
Perceive pleasure, Perceive curiosity.	Leverage the capabilities of the Metaverse and adopt a more attractive Metaverse banking design.
Perceptual personification	Adopt a more anthropomorphic design to highlight human features in the picture.
Perceived authenticity	Adopt more realistic, authentic bank images.
Low-cost usage threshold	Lower the usage threshold of the Metaverse banking to enable more users to participate.
Social norms	Establish social norms for the Metaverse banking to limit the negative impacts of free interaction among users.
Satisfaction and user innovation	Continuously follow up on the demands of Metaverse banking users and quickly improve and optimise.

Regulations should be established to guide user-to-user interactions and prevent potential negative consequences resulting from unregulated engagement, which could otherwise hinder user adoption and satisfaction [40].

During the early implementation of Metaverse banking [37], banks must consistently monitor and respond to customer needs, providing timely feedback to improve customer satisfaction quickly [21]. Banks should actively encourage users to raise concerns and iteratively refine the platform based on user feedback, ensuring a user-centred and responsive development process.

D. RQ4

The current implementation challenges of Metaverse banking are presented in Table VI. The primary distinction between Metaverse banking and traditional online banking lies in Metaverse banking's ability to extend users' social interactions and conduct banking activities within a virtual world. This dual functionality benefits banks and users and introduces greater uncertainty and risk. A more open and flexible Metaverse banking platform increases the risk of personal privacy breaches during user activities. Without clearly defined social norms, interactions between users may lead to virtual misconduct, such as verbal abuse or humiliation, compromising user safety. Establishing social norms within the Metaverse is a critical challenge during the early adoption phase of Metaverse banking [38].

The involvement of authoritative institutions in implementing Metaverse banking can further enhance users' perceived ease of use. Users are more likely to perceive Metaverse banking platforms supported by credible institutions as more straightforward. Such institutional backing also contributes to building social influence and a trustworthy brand image in the financial market. Establishing a reliable social influence and a credible brand image represents another major challenge during the initial adoption phase [40].

Users already have expectations regarding the processing speed of banking services. As such, they will likely express concerns about processing time and system performance when engaging in financial transactions [41]. During the early technological development stage, it is essential for Metaverse banking to ensure the timely completion of tasks and performance that meets user expectations to prevent apprehension. Ensuring stable system performance is key to enhancing users' perceived reliability of Metaverse banking technologies.

As Metaverse banking expands user interaction, it is necessary to determine how users should operate their virtual avatars and what methods can effectively facilitate user-to-user engagement [36].

In its early stages, Metaverse banking should avoid overly complex system designs that may overwhelm users with perceived technical difficulty [29]. Making the technology feel simple and easy to use remains one of the foremost challenges for successfully adopting Metaverse platforms.

TABLE VI. POTENTIAL CHALLENGES IN THE INITIAL DEVELOPMENT OF METAVERSE BANKING

Challenging factors	Factor Description
Privacy risks	The virtual world inevitably generates risks such as privacy leakage and behaviour monitoring.
Social norms	Improper behaviour between users will pose a threat to users.
Ease of use	How to make users feel that it is simple and easy to use
Performance expectations	How to avoid users' concerns about bank performance
Metaverse interactions	How to promote the more convenient use of avatars by Metaverse users

VI. CONCLUSION

This study primarily focuses on an exploratory literature review of the market adoption of Metaverse banking during its early development stage. It identifies the adoption characteristics of traditional banking users from the perspective of online banking. It integrates this with literature on service quality in Metaverse banking to address the current research gap regarding how Metaverse technologies influence user adoption in the early stages of development within the banking sector. The study reveals key features and needs of online banking users alongside the actual performance of Metaverse technologies at this stage of development. It offers implementation recommendations for Metaverse Banking and outlines future research directions, thereby accelerating the integration of Metaverse technologies into the banking industry.

The research emphasises that Metaverse Banking should construct a coherent Metaverse social framework, encompassing the operational mechanisms of virtual environments and social interaction mechanisms for banking users. More effective technological solutions should be adopted to ensure customer privacy and financial security. However, the current research findings remain limited, as this study has not yet conducted empirical validation through practical cases. Thus, the validity of these conclusions awaits further verification. Accordingly, from the perspective of the initial implementation of Metaverse Banking, this study provides a reference for technological implementation rather than a definitive and comprehensive guideline.

Based on the current research findings, future studies should adopt empirical approaches to further clarify the interrelationships among factors influencing users during the initial adoption stage. Particular attention should be given to understanding users' psychological states when they first engage with banking services in immersive interactive environments, thereby providing guidance for integrating Metaverse technologies into banking and facilitating the transformation of banking services in the new digital era.

ACKNOWLEDGMENT

This work was supported by the RHB-UKM Endowment Fund: RHB-UKM-2024-001 and Universiti Kebangsaan Malaysia (UKM) for the article publication financial support.

REFERENCES

[1] V. Dubey, A. Mokashi, R. Pradhan, P. Gupta, and R. Walimbe, 'Metaverse and Banking Industry – 2023 The Year of Metaverse Adoption', Technium: Romanian Journal of Applied Sciences and Technology, vol. 4, no. 10, pp. 62–73, Nov. 2022, doi: 10.47577/technium.v4i10.7774.

[2] A. Kumar and A. Shankar, 'Will metaverse revolutionize retail banking? Understanding user acceptance towards metaverse banking', International Journal of Retail & Distribution Management, vol. 52, no. 6, pp. 629–646, Jul. 2024, doi: 10.1108/IJRDM-06-2023-0396.

[3] R. Canales and R. Nanda, 'A darker side to decentralized banks: Market power and credit rationing in SME lending', J. Finan. Econ., vol. 105, no. 2, pp. 353–366, Aug. 2012, doi: 10.1016/j.jfineco.2012.03.006.

[4] K.-B. Ooi et al., 'Banking in the metaverse: a new frontier for financial institutions', International Journal of Bank Marketing, vol. 41, no. 7, pp. 1829–1846, Dec. 2023, doi: 10.1108/IJBM-03-2023-0168.

[5] M. Z. L. Zainurin, M. Haji Masri, M. H. A. Besar, and M. Anshari, 'Towards an understanding of metaverse banking: a conceptual paper', JFRA, vol. 21, no. 1, pp. 178–190, Mar. 2023, doi: 10.1108/JFRA-12-2021-0487.

[6] Y. Wang et al., 'A Survey on Metaverse: Fundamentals, Security, and Privacy', IEEE Commun. Surv. Tutorials, vol. 25, no. 1, pp. 319–352, 2023, doi: 10.1109/COMST.2022.3202047.

[7] Z. ÇelİK, B. Dülek, İ. Aydin, and R. Saydan, 'METAVERSE: BIBLIOMETRIC ANALYSIS, A CONCEPTUAL MODEL PROPOSAL, AND A MARKETING-ORIENTED APPROACH', Bingöl Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, no. 24, pp. 383–394, Oct. 2022, doi: 10.29029/busbed.1114777.

[8] A. Papathomas and G. Konteos, 'Financial institutions digital transformation: the stages of the journey and business metrics to follow', J Financ Serv Mark, vol. 29, no. 2, pp. 590–606, Jun. 2024, doi: 10.1057/s41264-023-00223-x.

[9] S. Chauhan, A. Akhtar, and A. Gupta, 'Customer experience in digital banking: a review and future research directions', IJQSS, vol. 14, no. 2, pp. 311–348, May 2022, doi: 10.1108/IJQSS-02-2021-0027.

[10] A. Koohang et al., 'Shaping the Metaverse into Reality: A Holistic Multidisciplinary Understanding of Opportunities, Challenges, and Avenues for Future Investigation', Journal of Computer Information Systems, vol. 63, no. 3, pp. 735–765, May 2023, doi: 10.1080/08874417.2023.2165197.

[11] E. A. Firmansyah and U. H. Umar, 'Metaverse in business research: a systematic literature review', Cogent Business & Management, vol. 10, no. 2, p. 2222499, Dec. 2023, doi: 10.1080/23311975.2023.2222499.

[12] M. Goldberg and F. Schär, 'Metaverse governance: An empirical analysis of voting within Decentralized Autonomous Organizations', Journal of Business Research, vol. 160, p. 113764, May 2023, doi: 10.1016/j.jbusres.2023.113764.

[13] 'Challenges in Distributed Agile Software Development Environment: A Systematic Literature Review', KSII TIIS, vol. 13, no. 9, Sep. 2019, doi: 10.3837/tiis.2019.09.013.

[14] H. Huawei et al., 'Economic Systems in the Metaverse: Basics, State of the Art, and Challenges', ACM Computing Surveys, vol. 56, no. 4, pp. 1–33, Apr. 2024, doi: 10.1145/3626315.

[15] V. Kandpal, P. K. Ozili, P. M. Jeyanthi, D. Ranjan, and D. Chandra, 'Applications of Metaverse Technology on Banking Services: Opportunities and Challenges', in Digital Finance and Metaverse in Banking, Emerald Publishing Limited, 2025, pp. 69–86. doi: 10.1108/978-1-83662-088-420251003.

[16] B. Mwiya et al., 'Examining the effects of electronic service quality on online banking customer satisfaction: Evidence from Zambia', Cogent Business and Management, vol. 9, no. 1, 2022, doi: 10.1080/23311975.2022.2143017.

[17] S. A. Raza, A. Umer, M. A. Qureshi, and A. S. Dahri, 'Internet banking service quality, e-customer satisfaction and loyalty: the modified e-SERVQUAL model', TQM, vol. 32, no. 6, pp. 1443–1466, Nov. 2020, doi: 10.1108/TQM-02-2020-0019.

[18] R. Reepu and R. Arora, 'The Effect of Perceived Risk on Intention to Use Online Banking', ujaf, vol. 10, no. 1, pp. 62–71, Jan. 2022, doi: 10.13189/ujaf.2022.100107.

[19] S. Kaur and S. Arora, 'Role of perceived risk in online banking and its impact on behavioral intention: trust as a moderator', Journal of Asia Business Studies, vol. 15, no. 1, pp. 1–30, 2021, doi: 10.1108/JABS-08-2019-0252.

[20] A. H. T. Pham, D. X. Pham, E. I. Thalassinos, and A. H. Le, 'The Application of Sem–Neural Network Method to Determine the Factors Affecting the Intention to Use Online Banking Services in Vietnam', Sustainability, vol. 14, no. 10, p. 6021, May 2022, doi: 10.3390/su14106021.

[21] M. B. Abdul Sathar, M. Rajagopalan, S. M. Naina, and S. Parayitam, 'A moderated-mediation model of perceived enjoyment, security and trust on customer satisfaction: evidence from banking industry in India', JABS, vol. 17, no. 3, pp. 656–679, Apr. 2023, doi: 10.1108/JABS-03-2022-0089.

[22] W. Nasri, 'Acceptance of Internet Banking in Tunisian Banks: Evidence From Modified UTAUT Model', International Journal of E-Business Research, vol. 17, no. 3, pp. 1–21, Jul. 2021, doi: 10.4018/IJEBR.2021070102.

[23] A. B. Abdelrhman, 'Assessing The Customers' Behavioral Intention to Adopt Internet Banking Services', Pak.j.life.soc.sci., vol. 22, no. 2, 2024, doi: 10.57239/PJLSS-2024-22.2.00719.

[24] M. Asif, M. A. Khan, H. Alhumoudi, and M. Wasiq, 'Examining the Role of Self-Reliance, Social Domination, Perceived Surveillance, and Customer Support with Respect to the Adoption of Mobile Banking', IJERPH, vol. 20, no. 5, p. 3854, Feb. 2023, doi: 10.3390/ijerph20053854.

[25] Y. Alshboul and N. Al Hamouri, 'Cybersecurity antecedents of trust: toward OPS adoption in Jordan', International Journal of Information and Decision Sciences, vol. 15, no. 1, pp. 73–93, 2023, doi: 10.1504/IJIDS.2023.129654.

[26] A. Akhter, Md. Mobarak Karim, S. Jannat, and K. M. Anwarul Islam, 'Determining factors of intention to adopt internet banking services: A study on commercial bank users in Bangladesh', Banks and Bank Systems, vol. 17, no. 1, pp. 125–136, Mar. 2022, doi: 10.21511/bbs.17(1).2022.11.

[27] S. Singh and R. K. Srivastava, 'Understanding the intention to use mobile banking by existing online banking customers: an empirical study', J Financ Serv Mark, vol. 25, no. 3–4, pp. 86–96, Dec. 2020, doi: 10.1057/s41264-020-00074-w.

[28] G. Albort-Morant, C. Sanchís-Pedregosa, and J. R. Paredes Paredes, 'Online banking adoption in Spanish cities and towns. Finding differences through TAM application', Economic Research-Ekonomska Istraživanja, vol. 35, no. 1, pp. 854–872, Dec. 2022, doi: 10.1080/1331677X.2021.1945477.

[29] A. Kumar and A. Shankar, 'Will metaverse revolutionize retail banking? Understanding user acceptance towards metaverse banking', IJRDM, vol. 52, no. 6, pp. 629–646, Jul. 2024, doi: 10.1108/IJRDM-06-2023-0396.

[30] B. Kaur and N. K. Batra, 'Technology adoption of digital banking and women consumers: An empirical investigation', IJERR, vol. 32, pp. 278–287, Aug. 2023, doi: 10.52756/ijerr.2023.v32.024.

[31] W.-R. Lin, Y.-H. Wang, and Y.-M. Hung, 'Analyzing the factors influencing adoption intention of internet banking: Applying DEMATEL-ANP-SEM approach', PLOS ONE, vol. 15, no. 2, p. e0227852, Feb. 2020, doi: 10.1371/journal.pone.0227852.

[32] S. Sharma, J. Singh, A. Gupta, F. Ali, F. Khan, and D. Kwak, 'User Safety and Security in the Metaverse: A Critical Review', IEEE Open J. Commun. Soc., vol. 5, pp. 5467–5487, 2024, doi: 10.1109/OJCOMS.2024.3397044.

[33] A. L. M. Anouze and A. S. Alamro, 'Factors affecting intention to use e-banking in Jordan', IJBM, vol. 38, no. 1, pp. 86–112, Jul. 2019, doi: 10.1108/IJBM-10-2018-0271.

[34] T. Jung et al., 'Metaverse for service industries: Future applications, opportunities, challenges and research directions', Computers in Human Behavior, vol. 151, p. 108039, Feb. 2024, doi: 10.1016/j.chb.2023.108039.

[35] A. Garepasha, S. Aali, A. R. Bafandeh Zendeh, and S. Iranzadeh, 'Relationship dynamics in customer loyalty to online banking services', JIMA, vol. 12, no. 4, pp. 830–863, May 2021, doi: 10.1108/JIMA-09-2019-0183.

[36] A. M. Aburbeian, A. Y. Owda, and M. Owda, 'A Technology Acceptance Model Survey of the Metaverse Prospects', AI, vol. 3, no. 2, pp. 285–302, Apr. 2022, doi: 10.3390/ai3020018.

[37] M. Al-kfairy, A. Alomari, M. Al-Bashayreh, O. Alfandi, and M. Tubishat, 'Unveiling the Metaverse: A survey of user perceptions and the impact of usability, social influence and interoperability', Heliyon, vol. 10, no. 10, p. e31413, May 2024, doi: 10.1016/j.heliyon.2024.e31413.

[38] G. Kang, J. Koo, and Y.-G. Kim, 'Security and Privacy Requirements for the Metaverse: A Metaverse Applications Perspective', IEEE Commun. Mag., vol. 62, no. 1, pp. 148–154, Jan. 2024, doi: 10.1109/MCOM.014.2200620.

[39] K. Shahzad, M. Ashfaq, A. U. Zafar, and S. Basahel, 'Is the future of the metaverse bleak or bright? Role of realism, facilitators, and inhibitors in metaverse adoption', Technological Forecasting and Social Change, vol. 209, p. 123768, Dec. 2024, doi: 10.1016/j.techfore.2024.123768.

[40] X. Bao, J. Yu, and M. Shou, 'Exploring metaverse: affordances and risks for potential users', ITP, Sep. 2024, doi: 10.1108/ITP-03-2023-0222.

[41] C. H. Heruatmadja and A. Ramadhan, 'Drivers and barriers of intention to work within metaverse environment', Multimed Tools Appl, vol. 83, no. 17, pp. 50955–50979, Nov. 2023, doi: 10.1007/s11042-023-17567-0.

Optimizing Image Retrieval: A Two-Step Content-Based Image Retrieval System Using Bag of Visual Words and Color Coherence Vectors

Muhammad Sauood[1], Muhammad Suzuri Hitam[2]*, Wan Nural Jawahir Hj Wan Yussof[3]
Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia [1, 2, 3]
Department of Computer Sciences, Bahria University Lahore Campus, Lahore, Pakistan [1]

***Abstract*—Content-Based Image Retrieval (CBIR) systems play a crucial role in efficiently managing and retrieving images from large datasets based on visual content. This paper presents a novel bi-layer CBIR system that integrates Bag of Visual Words (BoVW) and Color Coherence Vector (CCV) methods to enhance image retrieval accuracy and performance by leveraging the strengths of both feature extraction techniques. In the first layer, the BoVW approach extracts local features and represents images as histograms of visual word occurrences, facilitating efficient initial filtering. In the second layer, CCV features are extracted from the top retrieved images to capture the spatial coherence of colour regions, providing a detailed colour signature. By combining the merits of both layers, the proposed system achieves higher retrieval precision and recall compared to the traditional single-layer approaches. Experimental results demonstrate the effectiveness of the bi-layer CBIR system in retrieving relevant images with improved accuracy, making it a valuable tool for application in image databases, digital libraries, and multimedia content management.**

***Keywords*—CBIR system; Bag of Visual Words; color coherence vectors, bi-layer CBIR; two-step CBIR feature fusion; feature extraction**

I. Introduction

As image data grows, the process to retrieve similar images from a large database becomes very challenging. The traditional approach to search images is known as text-based image retrieval (TBIR), which uses a technique where images are annotated automatically or manually, which is expensive in terms of labor, and it is very time-consuming.

In TBIR system, image search is based on metadata of the query image which leads to some subjective issues, such as variation in human perception of the image, misleading annotations that lead to inaccurate image retrieval results [1].

CBIR was introduced in early 1980s to overcome the disadvantages of traditional approach in TBIR. In CBIR approach, instead of giving text as a query, an image is feed into the system as a query and the system will look into the image database to retrieve for similar look and relevant images [2]. CBIR is the answer by the computer vision for difficulty in retrieving images from huge databases. CBIR has many applications like facial recognition systems [3], in the field of military and defence [4], retail catalogues and photograph archives [5], medical imaging [6], intellectual property [7], fight against crime [8], detection of nude and inappropriate material [9] and collection of art [10].

In CBIR system, there are two major processes. The first process is feature extraction, and the second process is feature similarity matching. In the first process, low-level features such as color, texture and shape, were extracted from both the query and image from the database using an appropriate feature extraction method. In the second process, distance measurement metrics is used to measure the similarity between the features from the query image and the respective features from the image database. Retrieved images are ranked according to the lowest distance value. The smaller the distance value, the more similar the image is according to the used features [11]. Thus, it is critical and important to choose appropriate feature extraction method(s).

Content-based image retrieval methods that rely on a single feature often produce suboptimal performance [12][13] because they capture only a limited aspect of image content. Recent literature also shows that combining multiple descriptors can address these limitations [14][15].

This paper proposes a bi-layer CBIR system that combines two complementary feature extraction techniques: Bag of Visual Words (BoVW) [16] and Color Coherence Vector (CCV) [17]. The BoVW model, inspired by text retrieval methods, has been widely adopted for its ability to handle local features and represent images as histograms of visual word occurrences.

However, BoVW primarily focuses on local descriptors and often overlooks the global spatial coherence of colour regions. To address this limitation, the CCV method is incorporated in the second layer to capture the coherence of colour regions, providing a more holistic representation of image content. The proposed system operates in two layers. In the first layer, images are indexed using the BoVW model, which involves extracting local feature descriptors, constructing a visual vocabulary through k-means clustering, and representing each image as a histogram of visual word occurrences [16]. This process allows for efficient initial filtering of images. In the second layer, the top retrieved images undergo further analysis using CCV features, which quantify the coherence of colour regions within the images [17]. Integrating these two methods enhances the system's ability to discriminate between visually similar images, thereby improving retrieval accuracy.

*Corresponding author.

The remainder of this paper is organized as follows. Section II reviews related work on CBIR, with an emphasis on advances in global, local, and hybrid descriptors. Section III describes the proposed two-step CBIR methodology in detail. Section IV provides an overview of the Corel database used for evaluation, while Section V presents and analyzes the experimental results. Finally, Section VI offers concluding remarks and outlines potential directions for future research.

II. Related Work

A content-based image retrieval (CBIR) method that leverages Zernike chromaticity distribution moments in opponent chromaticity space for color features and employs rotation and scale-invariant Contourlet transform descriptors for texture analysis was introduced in [18]. This combined feature set is both compact and robust, with image similarity assessed through a weighted distance measure. General structure of the CBIR approach is shown in Fig. 1.

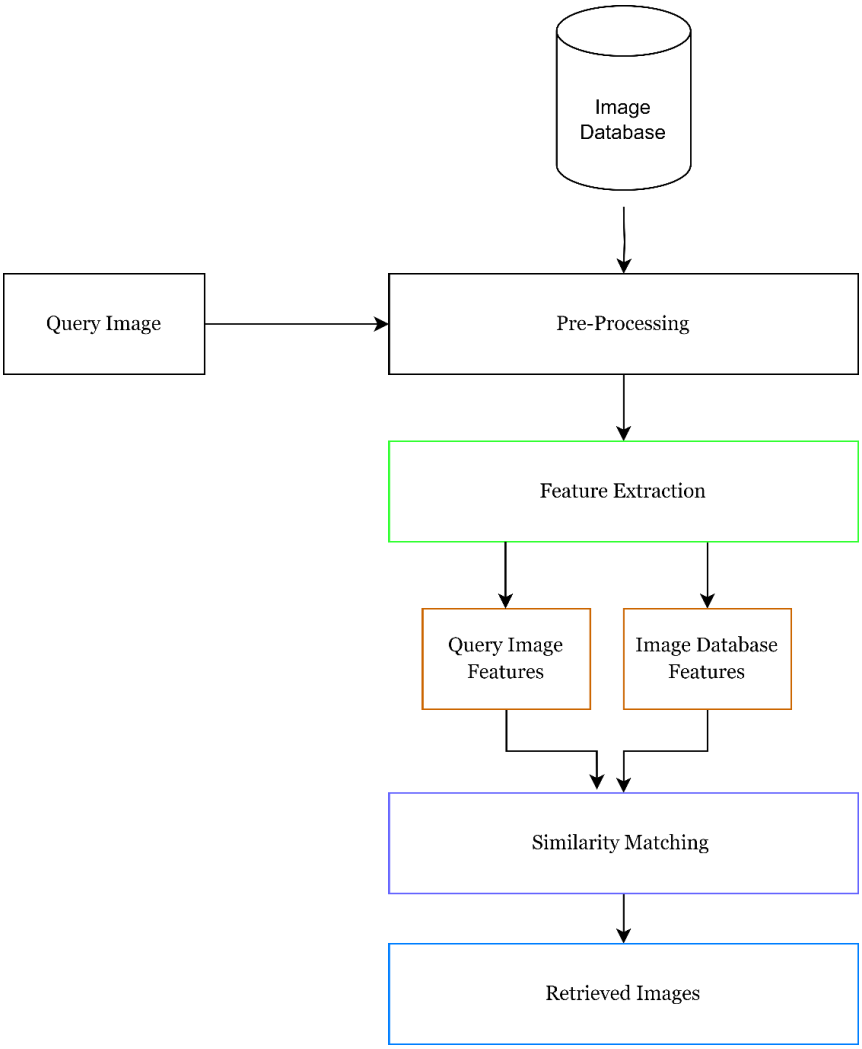


Fig. 1. General structure of the CBIR approach.

Experimental results on the COREL dataset indicate the approach achieves approximately 67% precision. The method demonstrates improved retrieval effectiveness by integrating both color and texture cues and offers resilience to variations in image brightness. However, the study provides limited insights regarding computational requirements and lacks an in-depth analysis of how parameter choices might influence retrieval outcomes. A CBIR method that fuses color histogram features with local directional pattern (LDP) descriptors to capture both color and texture characteristics of images was proposed in [19]. The approach incorporates feature normalization and a novel weighted distance metric for similarity measurement. Extensive experiments on the Wang and Corel-10000 datasets demonstrate the method's effectiveness, achieving a precision of 78.31% on the Wang dataset and a precision of 51.91% with a recall of 6.23% on the Corel-10000 dataset, outperforming several existing techniques. Strengths of this method include its robust fusion of complementary features and the clear improvement in retrieval accuracy. However, the retrieval system introduces a higher feature dimensionality, and computational efficiency is somewhat impacted compared to methods with lower-dimensional representations.

A CBIR system employing a hybrid clustering strategy that combines particle swarm optimization (PSO) with k-means clustering was presented in [20]. The method leverages color

histograms, color moments, co-occurrence matrices, and wavelet moments to extract color and texture features, and uses clustering to minimize search space and enhance retrieval speed. Experiments on the WANG dataset, which includes 1,000 images distributed across 10 classes, show the system achieves an average precision of 80.5% and a maximum class precision of 0.998 for dinosaurs, outperforming several existing methods.

Strengths of the system include its effective integration of clustering to improve retrieval accuracy and reduce computational cost at the retrieval stage. However, the system demonstrates lower precision for certain classes, such as buses and architecture, due to clustering errors, and does not incorporate shape features.

An efficient CBIR method for large databases utilizing a three-stage retrieval process based on color histograms, Gabor texture features, and Fourier descriptor-based shape features was proposed in [21]. In this approach, retrieval is performed sequentially: first selecting the top N images by color similarity, then refining to P images by texture similarity, and finally to K images using shape similarity, thereby removing the need for feature fusion and normalization. For the COREL 1k dataset, the method achieved an average precision of 76.9%. Key strengths include reduced computation time due to stage-wise narrowing of the search space and improved retrieval accuracy by integrating both global and region features.

However, the effectiveness of the system depends on careful selection of the N, P, and K parameters, and the reliability of shape features can be limited by segmentation quality, especially in complex natural images. A summary of the methods employing global features is presented in Table I.

TABLE I. COMPARISON OF CONTENT-BASED IMAGE RETRIEVAL METHODS USING GLOBAL FEATURES

Author (Year)	Key Features	Dataset	Precision	Strengths	Weaknesses
Wang et al. (2013)	Zernike chromaticity, Contourlet texture	COREL 1k	67.0%	Compact, robust, illumination-invariant	Lacks computational, parameter analysis
Zhou et al. (2018)	Color histogram + LDP	Wang, Corel-10k	78.3% (Wang), 51.9% (Corel-10k)	Strong fusion, high accuracy	Higher dimensionality, efficiency impact
Zeyad Safaa Younus et al. (2014)	PSO + k-means clustering, color & texture	WANG 1k	80.5%	Accurate, efficient clustering	Some classes are low, no shape features
Shrivastava & Tyagi (2015)	3-stage: color, texture, shape	COREL 1k	76.9%	Fast, integrates global & regional	Needs tuning, segmentation limits

A CBIR method utilizing local features by integrating scale-invariant feature transform (SIFT) and speeded up robust features (SURF) visual words within a bag-of-features framework was proposed in [22]. Separate codebooks are constructed for SIFT and SURF, and their concatenation yields a robust image representation, with similarity measured via a support vector machine (SVM) using a Hellinger kernel. Experiments on the Corel-1000, Corel-1500, and Corel-2000 datasets demonstrate mean average precision of 75.17%, 74.95%, and 65.41%, respectively, surpassing several state-of-the-art methods. The approach's main strengths are improved retrieval performance and robustness to scale, rotation, and illumination changes by combining both descriptors. However, the method increases feature dimensionality and relies on parameter choices (such as vocabulary size and clustering), whose effects on retrieval effectiveness and computational efficiency are not deeply analyzed.

A method for CBIR that combines Local Binary Pattern Variance (LBPV) and Local Intensity Order Pattern (LIOP) features was presented in [23]. By forming two compact visual dictionaries and then concatenating them, the method achieves robust representation and improved retrieval performance. Dimensionality reduction using principal component analysis (PCA) and the use of SVM for classification help manage computational load. The proposed system demonstrated high precision, 76.02% on WANG-1.5K, and was efficient in terms of computational cost. However, it is not directly applicable to multispectral images due to loss of spectral and spatial detail.

An encrypted image retrieval method for cloud computing that integrates optimized Harris corner detection with SURF features within a Bag-of-Words model and employs p-stable Locality Sensitive Hashing (LSH) for indexing was proposed in [24]. The approach extracts local feature points via an improved Harris algorithm, enhances feature description with SURF, and utilizes a chaotic encryption scheme to ensure the security of both images and indexes. Experimental evaluation on the Corel test set (1000 images, 10 categories) shows the proposed scheme achieves higher retrieval accuracy and lower search time compared to existing encrypted image retrieval methods, with a top-10 retrieval precision of up to 54.3%. Strengths of the method include increased retrieval efficiency and security through efficient local feature extraction and optimized hashing. However, retrieval precision decreases as the number of retrieved images increases, and the effectiveness of the scheme depends on parameter choices (e.g. number of hash tables, LSH functions), which are not deeply analyzed for sensitivity or scalability. Table II provides an overview of methods that utilize local features.

A CBIR system that integrates both local and global features by combining SIFT, Histogram of Oriented Gradients (HOG), and Local Binary Pattern (LBP) descriptors within a bag-of-features (BoF) framework was proposed in [25]. The approach explores both patch-based and image-based integration models, with feature codebooks constructed using a weighted K-means clustering algorithm. Experiments on the Corel-1000 dataset show that the image-based SIFT-LBP integration achieves the highest average retrieval precision of 65.7%, outperforming conventional color, texture, and shape-based approaches as well as BoF and spatial pyramid matching (SPM) baselines. Key strengths include complementary representation of image content and superior robustness against background noise. However, the system introduces a higher computational cost due to combined feature extraction, and its performance is sensitive to codebook size and feature weighting parameters, which require careful tuning.

TABLE II. COMPARISON OF CONTENT-BASED IMAGE RETRIEVAL METHODS USING LOCAL FEATURES

Author (Year)	Key Features / Approach	Dataset(s)	Precision (MAP/Top-10)	Strengths	Weaknesses
Nouman Ali et al. (2016)	SIFT + SURF (BoW), dual codebooks, SVM-Hellinger	Corel-1000/1500/2000	75.17% / 74.95% / 65.41%	High retrieval, robust to scale/rotation/lighting	High dimensionality, parameter sensitivity
Sarwar et al. (2019)	LBPV + LIOP, dual dictionaries, PCA, SVM	WANG-1.5K	76.02%	Compact, efficient, improved accuracy	Not suited for multispectral images
Jiaohua Qin et al. (2019)	Harris + SURF (BoW), p-stable LSH, chaotic encryption	Corel-1k	54.3% (Top-10)	Secure, efficient hashing, improved accuracy	Precision drops for more retrieved images, parameter choices not deeply analyzed

A multi-level colored directional motif histogram (MLCDMH) approach for CBIR that integrates both color (global) and local structural features was introduced in [26]. The method extracts directional motif patterns at multiple levels, applies a weighted neighboring similarity scheme, and fuses directional features into a single invariant feature map, generating compact histograms from all color planes. On the Corel-1000 dataset, the system achieved an average precision of 64.00% for the top 10 retrieved images and 59.60% for the top 20, surpassing several motif- and non-motif-based approaches. Strengths include robust representation of both local and global image characteristics and reduced matching overhead through single-vector fusion. However, system performance is dependent on block size and motif pattern parameters, whose sensitivity is not extensively explored.

A CBIR system that integrates Hue Saturation and Value (HSV) color histograms (global color), discrete wavelet transform (global texture), and edge histogram descriptor (local texture) features was proposed in [27]. The method forms a composite feature vector from these three descriptors and measures similarity using Manhattan distance. Evaluation on the Corel-1k dataset shows that the proposed system achieves an average precision of 73.5% for the top-20 retrieved images, outperforming several state-of-the-art CBIR methods. Key strengths include the effective fusion of color and texture features, which enhances retrieval performance, and competitive results across diverse categories. However, the approach's performance is dependent on the choice of block sizes and quantization levels, and retrieval efficiency may be affected by increased feature vector dimensionality.

A CBIR method that combines local features SURF and global features HOG using both feature fusion and visual words fusion strategies within a Bag-of-Visual-Words (BoVW) framework was proposed in [28]. The approach constructs separate vocabularies for SURF and HOG, then fuses them to form a compact and robust image representation. Experiments on the Corel-1000, Corel-1500, and Corel-5K datasets show that the visual words fusion method achieves a mean average precision of 80.61%, 76.20%, and 60.60%, respectively, outperforming state-of-the-art techniques. Strengths include significantly improved retrieval performance and reduced semantic gap through integrated representation of local and global features. However, increased vocabulary size and feature dimensionality can lead to higher computational cost, and the method's performance is sensitive to codebook size and feature selection percentage, requiring careful tuning. Table III summarizes the comparison of CBIR methods integrating local and global features.

TABLE III. COMPARISON OF CONTENT-BASED IMAGE RETRIEVAL METHODS USING HYBRID LOCAL AND GLOBAL FEATURES

Author (Year)	Key Features / Approach	Dataset(s)	Precision (MAP)	Strengths	Weaknesses
Jing Yu et al. (2013)	SIFT, HOG, LBP (BoF), patch/image-based fusion	Corel-1000	65.7% (image-based SIFT-LBP)	Complementary, robust to noise	High computation; sensitive to codebook/weights
Jitesh Pradhan et al. (2020)	MLCDMH: color & local motif fusion	Corel-1000	64.0% (top 10), 59.6% (top 20)	Robust global/local fusion; compact representation	Sensitive to block/motif size, not explored
Atif Nazir et al. (2018)	HSV hist., DWT (global), Edge hist. (local)	Corel-1k	73.5% (top 20)	Effective color/texture fusion, strong category results	Block/quantization choice, high dimensionality
Zahid Mehmood et al. (2018)	SURF (local) + HOG (global), BoVW fusion	Corel-1000/1500/5K	80.6%, 76.2%, 60.6%	High performance; reduced semantic gap	High computation; sensitive to parameters

While CBIR approaches in recent literature utilize either global features, local features or combination of both local and global, a critical analysis reveals distinct strengths and limitations in these individual methods, which directly inform the rationale for combining Bag of Visual Words (BoVW) and Color Coherence Vector (CCV) in the proposed system.

The BoVW model offers significant advantages in large-scale CBIR tasks. BoVW provides a fixed-length feature vector regardless of image dimensions or complexity, enabling efficient indexing and fast retrieval in large image databases. It leverages local descriptors that are robust to scale, rotation, and illumination changes, making it suitable for scenarios involving partial occlusion or background clutter, as local features can capture salient regions even when portions of an object are missing or obscured. Additionally, the construction of the visual vocabulary in BoVW is typically unsupervised and does not require manual annotation, as k-means clustering is applied to local feature descriptors to build the codebook. However, BoVW predominantly focuses on the distribution of local features and often neglects the broader spatial and color coherence information within the image, potentially leading to suboptimal retrieval when global color structures play a significant role in image similarity.

In contrast, the Color Coherence Vector (CCV) method excels at capturing the spatial and texture structure of images by distinguishing between coherent and incoherent pixels.

Coherent pixels are part of large, connected regions of similar colors, enabling CCV to robustly represent objects that share dominant color regions, while filtering out scattered, insignificant color areas. CCV thus complements BoVW by capturing global color and spatial coherence—information often missed by purely local approaches.

Therefore, the combination of BoVW and CCV directly addresses the limitations of each individual method. By integrating BoVW's robust local feature representation with CCV's global color and spatial coherence descriptors, the proposed CBIR system can better discriminate between images that are similar in both structure and content, even in the presence of occlusions, clutter, or variable lighting conditions. This dual-layer approach is expected to improve retrieval precision, particularly for complex or diverse image datasets, and provides a sound justification for introducing both methods in the system.

III. Proposed Methodology

In this work, we propose a two-layer Content-Based Image Retrieval (CBIR) system that integrates local and global feature analysis in a cascaded manner to improve retrieval effectiveness. The system consists of an initial retrieval layer based on the Bag of Visual Words (BoVW) model using SURF descriptors, followed by a refinement layer utilizing Color Coherence Vector (CCV) features. Both retrieval layers employ the Manhattan (City-Block) distance metric for similarity measurement to ensure consistency and interpretability throughout the retrieval process.

A. Layer 1: Initial Retrieval via Bag of Visual Words (BoVW)

Let the image database be denoted as $\mathcal{D} = \{I_1, I_2, \dots, I_N\}$ and the query image as I_Q.

Step 1: Local Feature Extraction

For each image I, local descriptors are extracted using the SURF algorithm. Let the set of descriptors for image I be $\{d_1, d_2, \dots, d_N\}$, where each $d_i \in \mathbb{R}^D$ is a D-dimensional SURF feature vector.

Step 2: Visual Vocabulary Construction

All descriptors from the entire database are pooled and clustered via k-means clustering into K clusters, where each cluster center v_k $(k = 1,2, \dots, K)$ defines a visual word. The value of K is a design parameter.

Step 3: Histogram Encoding

Each descriptor d_i from image I is assigned to the nearest cluster center (visual word) based on the Euclidean norm. The BoVW histogram for image I is defined as:

$$h_k = \sum_{i=1}^{N_I} \delta\left(\arg\min_j |d_i - v_j|^2 = k\right) \quad (1)$$

where $\delta(\cdot)$ is the Kronecker delta function which equals 1 if the condition is true and 0 otherwise.

Step 4: Similarity Computation

The query image I_Q is similarly encoded as a BoVW histogram h_Q. The similarity between h_Q and each database histogram h_I is computed using the Manhattan distance:

$$D_1(h_Q, h_I) = \sum_{k=1}^{K} |h_{Q,k} - h_{I,k}| \quad (2)$$

where $h_{Q,k}$ and $h_{I,k}$ are the k^{th} bins of the query and candidate histograms, respectively.

Step 5: Candidate Selection

The images in the database are ranked in ascending order of $D_1(h_Q, h_I)$, and the top N_{top} candidates with the smallest distances (where $N_{top} = 50$ in our implementation) are selected for the second retrieval layer.

B. Layer 2: Refinement via Color Coherence Vector (CCV)

Step 1: Color Quantization

For each candidate image (from Layer 1) and the query image, color quantization is performed to map all pixels into L discrete color levels. Let the color level of pixel i be $c_i \in \{1,2, \dots, L\}$.

Connected component analysis is performed on each quantized image to identify spatially contiguous regions of pixels with the same color level. For each color level λ, pixels are partitioned into two categories based on the size of the connected region (component) they belong to:

- Coherent pixels: Pixels that belong to a connected component (region) whose size is greater than or equal to a specified threshold T.
- Non-coherent pixels: Pixels that belong to a connected component whose size is less than T.
- Formally, for each color level λ, the numbers of coherent and non-coherent pixels are computed as:

$$C_\lambda = \sum_{i \in coh} \delta(c_i = \lambda), N_\lambda = \sum_{i \in non-coh} \delta(c_i = \lambda) \quad (3)$$

where, C_λ and N_λ are the counts of coherent and non-coherent pixels at color level λ, respectively.

Step 2: Feature Vector Formation

The CCV feature vector $v = [C_1, N_1, C_2, N_2, \dots, C_L, N_L]$ of length $2L$ is constructed for each image.

Step 3: Similarity Computation

The similarity between the CCV feature vector of the query image v_Q and each candidate image v_I is computed using the Manhattan distance:

$$D_2(v_Q, v_I) = \sum_{j=1}^{2L} |v_{Q,j} - v_{I,j}| \quad (4)$$

where, $v_{Q,j}$ and $v_{I,j}$ are the j^{th} elements of the query and candidate CCV vectors, respectively.

Step 4: Final Ranking

The candidate images are ranked based on their $D_2(v_Q, v_I)$ values in ascending order. The retrieval output is the set of top 10 images most similar to the query in terms of color coherence.

The workflow of the proposed approach is shown in Fig. 2.

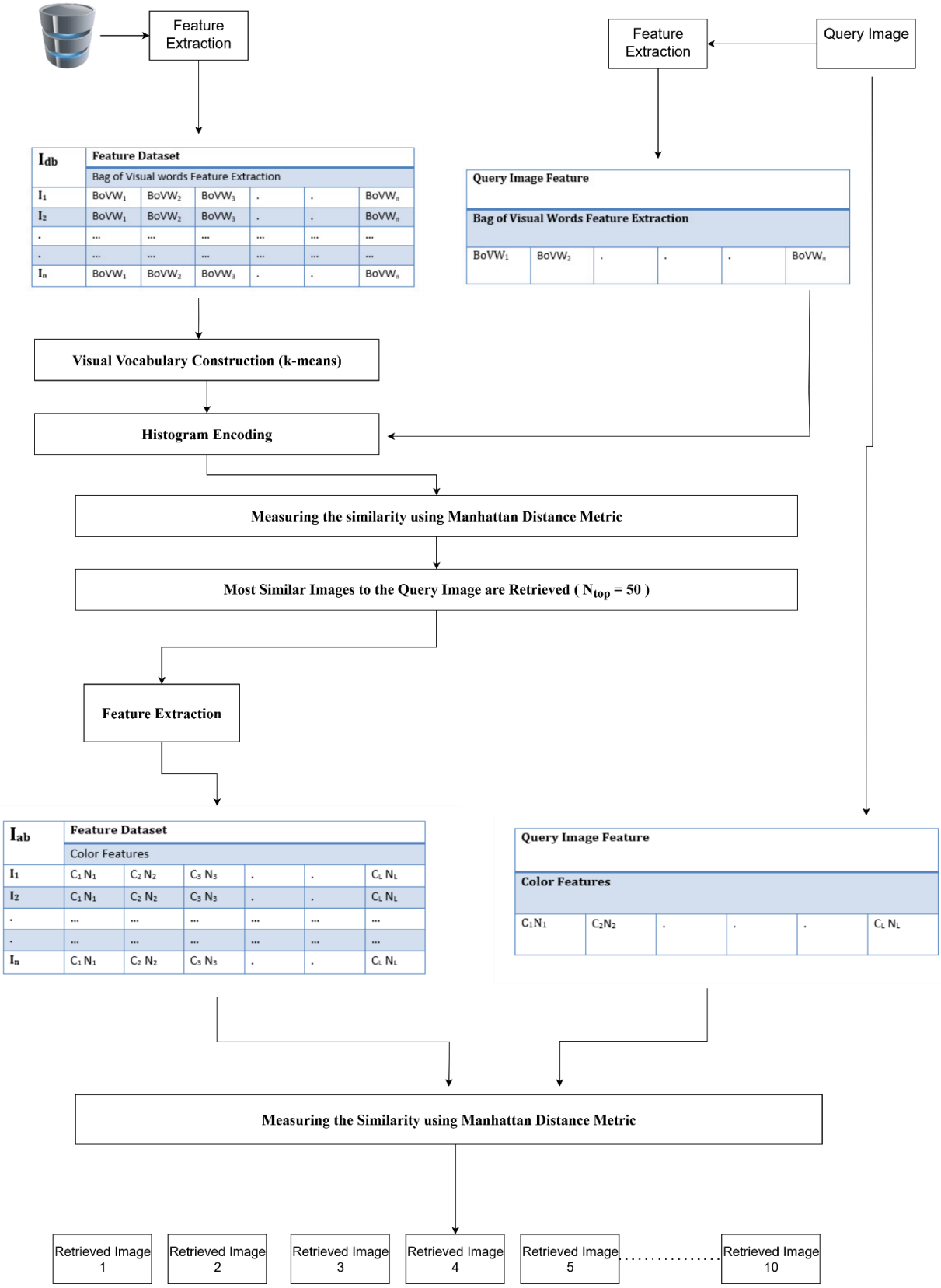


Fig. 2. Diagram of the proposed two step CBIR system.

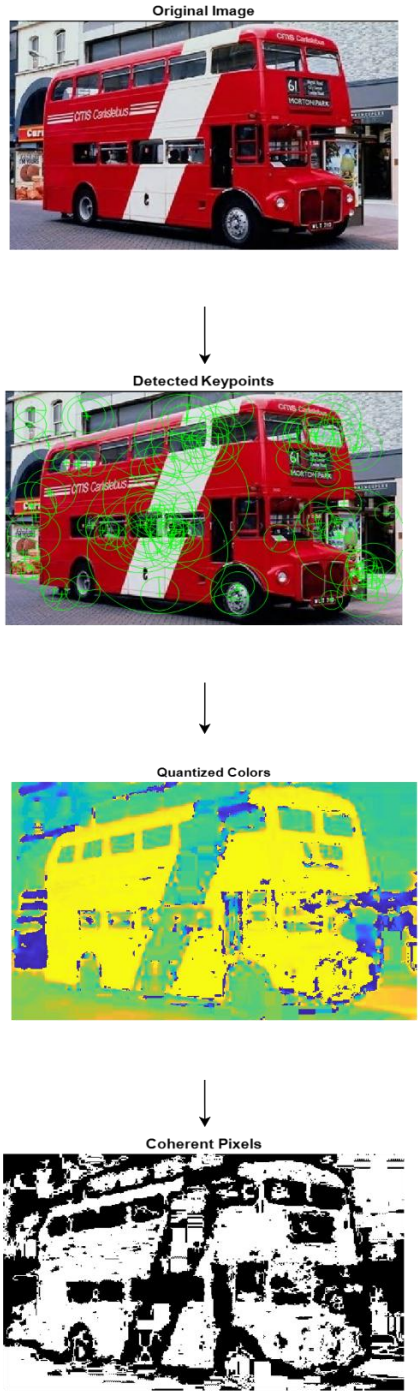


Fig. 3. Intermediate outputs of the proposed two-layer CBIR system.

Fig. 3 presents intermediate outputs from the proposed bi-layer CBIR framework. The first view shows the query image; the second depicts SURF keypoints used in the Bag-of-Visual-Words layer, which retrieves the Top-50 candidate images. The next view illustrates the Color Coherence Vector (CCV) computed in the second layer on these shortlisted images, followed by its binary coherence mask. Together, these steps show how the framework first filters candidates using local descriptors and then refines similarity using global color information.

IV. Database

The experiments are carried out on a publicly available database known as Corel-1k, which consists of 1,000 images organized into 10 categories. Every category consists of 100 images. The categories include natural scenes, everyday objects, animals, drinks, and aviation, among others. The database was selected because of its diversity and comprehensiveness. Corel-1k is a widely used benchmark image database in CBIR research, serving as a standard for evaluating retrieval accuracy and robustness. Each category in the database corresponds to different semantic concepts, facilitating the assessment of CBIR system performance. Some of the images are shown in Fig. 4 from the image database [29].

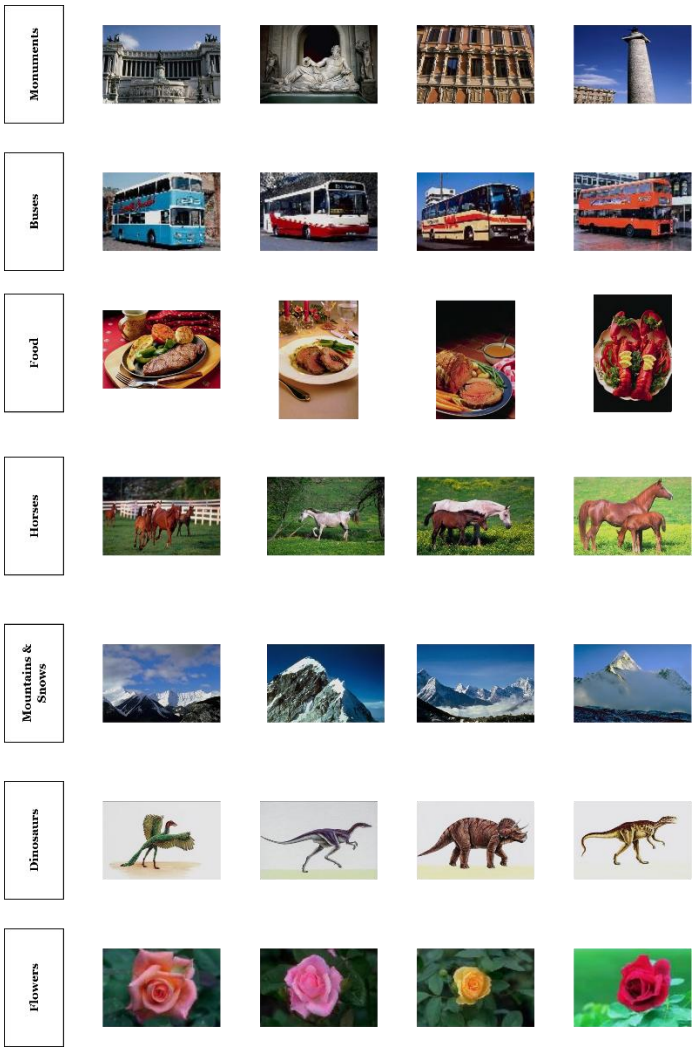


Fig. 4. Sample of images from Corel-1k dataset.

V. Experimental Results

This section evaluates the performance of the proposed novel two-layered CBIR approach. The system was assessed using the Corel-1K dataset, a benchmark frequently employed for testing CBIR methodologies due to its diverse content across

various categories. The performance metrics selected for evaluation were precision and recall, as these measures effectively indicate the system's accuracy and retrieval effectiveness.

Precision and recall are computed using standard formulas as follows:

Precision = (Number of relevant images retrieved) / (Total images retrieved).

Recall = (Number of relevant images retrieved) / (Total relevant images in the dataset).

Table IV presents the category-wise Precision@10 and Recall@10 (%) for the proposed two-layer CBIR system. The method attains an average precision of 82.2 % and a recall of 8.2 %, demonstrating that retrieved images are largely relevant even though recall is inherently limited by the top-10 evaluation protocol.

Strengths. Categories such as Buses, Dinosaurs, and Flowers achieve perfect precision, confirming that the approach effectively handles classes with uniform appearance and well-defined structure. Good results are also obtained for Horses and Elephants (P@10 = 90 %), highlighting the method's ability to manage moderate intra-class variation.

Weaknesses. Retrieval is less accurate for *Beaches* (P@10 = 45 %) and, to a lesser extent, Monuments *and* Mountains and Snow. These scene-type categories exhibit large variability in layout and background, leading to feature overlap with other classes.

Overall, the findings show that the proposed system excels in high-precision retrieval for object centric images, while scene categories remain challenging a direction for subsequent improvement.

Precision values demonstrate that most images retrieved at the top ranks are relevant. The comparatively small recall values arise from the evaluation strategy: since only the top 10 images are considered and each Corel-1K class contains 100 relevant images, even a perfect system can obtain at most 10 % recall.

The retrieval effectiveness of various CBIR methods was assessed primarily using the P@10 precision metric. The proposed approach achieved a precision rate of 82.2%, surpassing all benchmarked state-of-the-art methods considered in this comparative analysis as shown in Table V.

TABLE IV. PRECISION AND RECALL RATES FOR THE PROPOSED TWO-LAYERED CBIR SYSTEM

Categories	Precision @ 10	Recall @ 10
Buses	100	10
Dinosaurs	100	10
Horses	90	9
Elephants	90	9
Flower	100	10
Monuments	60	6
Mountains and Snow	70	7
Beaches	45	4.5
Africa	84	8.4
Food	83	8.3
Average	82.2	8.2

TABLE V. COMPARISON OF CBIR SYSTEMS IN TERMS OF PRECISION, DATASET, AND FEATURE EXTRACTION METHODS

Author(s) & Year	Features Used	Technique/Model	Distance Metric	Precision Rate
Lin et al. (2009)	CCM (color), DBPSP (texture), CHKM (color distribution)	Sequential Forward Selection (SFS)	Euclidean	72.70%
Huang et al. (2010)	Gabor texture, HSV color moment	Feature fusion	Euclidean	63.6%
Singha et al. (2012)	Color histogram, Wavelet transform	WBCHIR	Histogram Intersection	76.2%
Yu et al. (2013)	SIFT+LBP, HOG+LBP	Patch-based & image-based BoF integration	Weighted K-means	65%
Aiswarya et al. (2020)	Deep features via stacked autoencoder	Query image space generation + autoencoder	Not specified	67%
XIE et al. (2020)	Dominant color descriptor, Hu moments	Texton template-based zone detection	Not specified	80.40%
Nazir et al. (2018)	Edge histogram (local), DWT, color histogram (global)	Feature concatenation	Manhattan	73.5%
Sadique et al. (2019)	SURF, color moments (local), modified GLCM (global)	Feature concatenation	Approx. Nearest Neighbor	70.48%
Pradhan et al. (2019)	MLCDMH (local structural)	Multi-level histogram descriptor	Not specified	64%
Proposed Approach	Bag of Visual Words, Color Coherence Vector	Two-Step	Manhattan	82.2%

Among competing techniques, very few managed precision scores exceeding the 75% threshold, with the nearest competitor reaching 80.40%, and other notable methods clustered around 75–77%. Despite being close in performance, these methods ultimately fell short compared to the proposed system, highlighting the significant benefit of integrating local and global feature descriptors within a coherent retrieval framework.

Several other evaluated approaches exhibited moderate precision rates, generally between 65% and 74%, which encompassed both traditional feature engineering methods and hybrid techniques. Despite utilizing varied feature descriptors, these mid-tier methods demonstrated limitations either in the comprehensiveness of feature representation or in the consistency of similarity measurement, preventing them from achieving higher retrieval accuracy.

Methods at the lower end of performance attained precision scores below 65%, with the lowest being precisely 65%. This clearly indicates inadequate discriminative capability, likely attributable to limited descriptor diversity or suboptimal distance metrics. Although these methods frequently incorporated spatial and color information, their relatively weak retrieval results underline the necessity of employing more robust and complementary feature integration strategies.

The evident superiority of the proposed method, in terms of both absolute precision and the comparative performance margin, underscores the effectiveness of its two-phase retrieval strategy. This strategy, characterized by combining Bag of Visual Words (BoVW) and Color Coherence Vector (CCV), produces a synergistic enhancement of retrieval performance across diverse queries. The consistently superior performance relative to other benchmark methods confirms the robustness and practical applicability of the proposed CBIR approach in real-world scenarios.

To provide deeper insight into the retrieval capability and limitations of the proposed CBIR method, representative examples from two distinct categories “Dinasaurs” (high-performing) and 'Beach' (low-performing)—are presented. The precision-recall curves illustrated in Fig. 5 and Fig. 6 clearly highlight the method's superior performance for structured, visually coherent object categories, exemplified by the near-perfect retrieval results for the “Dinasaurs” class.

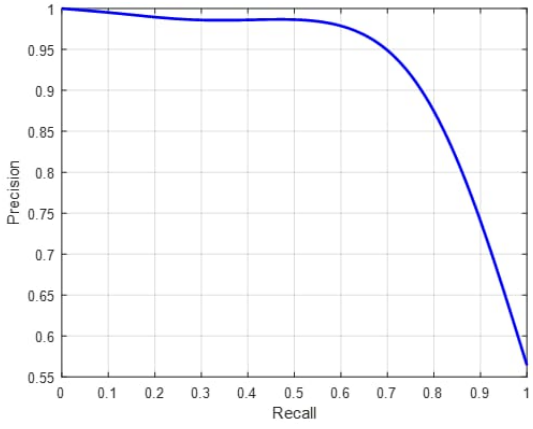


Fig. 5. Precision-recall graph for Dinasaurs class.

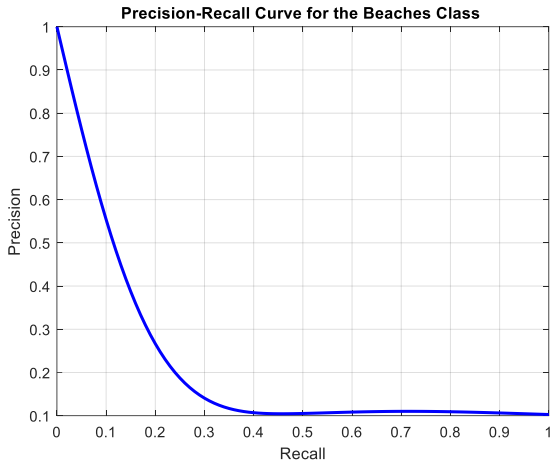


Fig. 6. Precision-recall graph for Beaches class.

Conversely, the “Beach” category exemplifies challenges inherent to semantically diverse and visually heterogeneous scenes, thus offering transparency into areas for further methodological refinement.

VI. Conclusion

This study aimed to advance the performance of CBIR systems by introducing a two-stage retrieval approach that effectively integrates local and global visual descriptors. Experimental evaluations demonstrated that the proposed framework performs competitively relative to existing state-of-the-art techniques, exhibiting improved accuracy in retrieving relevant images across diverse datasets.

By employing the Bag of Visual Words (BoVW) and Color Coherence Vector (CCV) descriptors, the proposed system successfully captures both structural and color-based visual information, significantly enhancing retrieval precision. The dual-layer architecture facilitates refined filtering and contributes positively to overall retrieval effectiveness without compromising generalization capabilities.

Despite these advantages, the method faces challenges regarding computational costs, particularly for high-resolution images and large-scale databases, as feature extraction and similarity matching become resource intensive. Future work will focus on designing lightweight variants of the framework, adopting more efficient feature-learning strategies and implementing real-time retrieval mechanisms to support interactive applications. Additional research may also explore relevance-feedback techniques and hybrid descriptors to further improve recall.

In conclusion, the developed CBIR framework offers a promising solution for practical applications, particularly in fields such as medical imaging, multimedia databases, and digital library management. Future optimizations may further enhance scalability and real-time responsiveness, positioning this methodology as a robust solution for evolving image retrieval challenges.

REFERENCES

[1] Rui, Yong, Thomas S. Huang, and Shih-Fu Chang. "Image retrieval: Current techniques, promising directions, and open issues." Journal of visual communication and image representation 10, no. 1 (1999): 39-62. https://doi.org/10.1006/jvci.1999.0413.

[2] Liu, Ying, Dengsheng Zhang, Guojun Lu, and Wei-Ying Ma. "A survey of content-based image retrieval with high-level semantics." Pattern recognition 40, no. 1 (2007): 262-282. https://doi.org/10.1016/j.patcog.2006.04.045

[3] Li Qinjun, Cui Tianwei, Zhao Yan, Wu Yuying, "Facial Recognition Technology: A Comprehensive Overview," Academic Journal of Computing & Information Science, Vol. 6, Issue 7, pp. 15-26, 2023, Francis Academic Press, UK. DOI: 10.25236/AJCIS.2023.060703.

[4] Kakizaki, K., Fukuchi, K., & Sakuma, J. (2023). Certified Defense for Content Based Image Retrieval. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) (pp. 4561-4570). IEEE.

[5] Lew, M. S., Sebe, N., Djeraba, C., & Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. ACM Transactions on Multimedia Computing, Communications, and Applications, 2(1), 1-19.

[6] Sudhish, D. K., Nair, L. R., & Shailesh, S. (2024). Content-based image retrieval for medical diagnosis using fuzzy clustering and deep learning. Biomedical Signal Processing and Control, 88, 105620. https://doi.org/10.1016/j.bspc.2023.105620

[7] António, J., Valente, J., Mora, C., Almeida, A., & Jardim, S. (2024). DarwinGSE: Towards better image retrieval systems for intellectual property datasets. PLOS ONE, 19(7), e0304915. https://doi.org/10.1371/journal.pone.0304915

[8] Liu, Y., Zhou, A., Xue, J., & Xu, Z. (2024). A Survey of Crime Scene Investigation Image Retrieval Using Deep Learning. Journal of Beijing Institute of Technology (English Edition), 33(4), 271-286. DOI: 10.15918/j.jbit1004-0579.2023.152

[9] Li, Xinfeng; Yang, Yuchen; Deng, Jiangyi; Yan, Chen; Chen, Yanjiao; Ji, Xiaoyu; Xu, Wenyuan. SafeGen: Mitigating Sexually Explicit Content Generation in Text-to-Image Models. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS '24), Salt Lake City, UT, USA, October 14-18, 2024. DOI: 10.1145/3658644.3670295.

[10] Castellano, G., Lella, E., & Vessio, G. (2021). Visual link retrieval and knowledge discovery in painting datasets. Multimedia Tools and Applications, 80, 6599–6616. https://doi.org/10.1007/s11042-020-09995-z

[11] Atlam, Hany Fathy, Gamal Attiya, and Nawal El-Fishawy. "Comparative study on CBIR based on color feature." International Journal of Computer Applications 78, no. 16 (2013).

[12] Al-Jubouri, Hanan, and Hongbo Du. "A Content-Based Image Retrieval Method By Exploiting Cluster Shapes." Iraqi Journal for Electrical & Electronic Engineering 14, no. 2 (2018).

[13] Sadique, Md Farhan, Bishajit Kumar Biswas, and SM Rafizul Haque. "Unsupervised content-based image retrieval technique using global and local features." In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), pp. 1-6. IEEE, 2019.

[14] Ashraf, Rehan, Mudassar Ahmed, Usman Ahmad, Muhammad Asif Habib, Sohail Jabbar, and Kashif Naseer. "MDCBIR-MF: multimedia data for content-based image retrieval by using multiple features." Multimedia tools and applications 79, no. 13 (2020): 8553-8579. https://doi.org/10.1007/s11042-018-5961-1

[15] Xie, Guangyi, Baolong Guo, Zhe Huang, Yan Zheng, and Yunyi Yan. "Combination of dominant color descriptor and Hu moments in consistent zone for content based image retrieval." IEEE Access 8 (2020): 146284-146299 10.1109/ACCESS.2020.3015285.

[16] Sivic, and Zisserman. "Video Google: A text retrieval approach to object matching in videos." In Proceedings ninth IEEE international conference on computer vision, pp. 1470-1477. IEEE, 2003.

[17] Pass, Greg, Ramin Zabih, and Justin Miller. "Comparing images using color coherence vectors." In Proceedings of the fourth ACM international conference on Multimedia, pp. 65-73. 1997.

[18] Wang, Xiang-Yang, Hong-Ying Yang, and Dong-Ming Li. "A new content-based image retrieval technique using color and texture information." *Computers & Electrical Engineering*, 39.3 (2013): 746-761.

[19] Zhou, Ju-Xiang, et al. "A new fusion approach for content based image retrieval with color histogram and local directional pattern." *International Journal of Machine Learning and Cybernetics*, 9 (2018): 677-689.

[20] Younus, Z. S., Mohamad, D., Saba, T., Alkawaz, H. M., Rehman, A., Al-Rodhaan, M., & Al-Dhelaan, A. "Content-based image retrieval using PSO and k-means clustering algorithm." *Arabian Journal of Geosciences*, 8(8), 6211–6224, 2015. doi:10.1007/s12517-014-1584-7

[21] Shrivastava, N., & Tyagi, V. "An efficient technique for retrieval of color images in large databases." *Computers & Electrical Engineering*, 46, 314–327, 2015. https://doi.org/10.1016/j.compeleceng.2014.11.009

[22] Ali, N., Bajwa, K. B., Sablatnig, R., Chatzichristofis, S. A., Iqbal, Z., Rashid, M., & Habib, H. A. "A novel image retrieval based on visual words integration of SIFT and SURF." *PloS one*, 11(6), e0157428, 2016.

[23] Sarwar, A., Mehmood, Z., Saba, T., Qazi, K. A., Adnan, A., & Jamal, H. "A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine." *Journal of Information Science*, 45 (1), 117–135, 2019. https://doi.org/10.1177/0165551518782825

[24] Qin, J., Li, H., Xiang, X., Tan, Y., Pan, W., Ma, W., & Xiong, N. N. "An encrypted image retrieval method based on harris corner optimization and LSH in cloud computing." *IEEE Access*, 7, 24626–24633, 2019. https://doi.org/10.1109/ACCESS.2019.2894673

[25] Yu, Jing, et al. "Feature integration analysis of bag-of-features model for image retrieval." *Neurocomputing*, 120 (2013): 355-364.

[26] Pradhan, Jitesh, et al. "Multi-level colored directional motif histograms for content-based image retrieval." *The Visual Computer*, 36.9 (2020): 1847-1868.

[27] Nazir, Atif, et al. "Content based image retrieval system by using HSV color histogram, discrete wavelet transform and edge histogram descriptor." *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*. IEEE, 2018.

[28] Mehmood, Z., Abbas, F., Mahmood, T., Javid, M. A., Rehman, A., & Nawaz, T. "Content-based image retrieval based on visual words fusion versus features fusion of local and global features." *Arabian Journal for Science and Engineering*, 43(12), 7265–7284, 2018. https://doi.org/10.1007/s13369-018-3062-0

[29] Duygulu, Pinar, Kobus Barnard, Joao FG de Freitas, and David A. Forsyth. "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary." In Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part IV 7, pp. 97-112. Springer Berlin Heidelberg, 2002.

An Aggregated Dataset of Agile User Stories and Use Case Taxonomy for AI-Driven Research

Abdulrahim Alhaizaey[1], Majed Al-Mashari[2]
Information Systems Department, King Khalid University, Abha, Saudi Arabia[1]
Information Systems Department, King Saud University, Riyadh, Saudi Arabia[1, 2]

***Abstract*—Agile methodologies are considered revolutionary approaches in the development of systems and software. With the rapid advancement of artificial intelligence, natural language processing, and large language models, there is an increasing demand for high-quality datasets to support the design and development of intelligent, practical, and effective automation tools. However, researchers in Agile Requirements Engineering face significant challenges due to the limited availability of datasets, particularly those involving user stories. This paper presents a dataset of over 10K user stories collected from academic sources and publicly accessible online repositories. These stories represent requirements formulated in accordance with Agile principles. The process of collecting and classifying data, as well as its use in a prior research project focused on identifying non-functional requirements, is described. The dataset was validated with substantial inter-annotator agreement and has been successfully employed in prior experiments, where a fine-tuned pre-trained language model achieved F1 scores above 93% in classifying non-functional requirements. Additionally, a structured taxonomy of potential research and practical use cases for this dataset is proposed, aiming to support researchers and practitioners in areas such as requirements analysis, automated generative tasks using generative language models, model development, and educational purposes.**

***Keywords*—Agile software development; requirements engineering; user stories; natural language processing; datasets; large language models; generative language models**

I. Introduction

Agile development methodologies have seen widespread adoption in recent years, leading to the prominent use of user stories as a primary tool for expressing software requirements [1], [2]. Lucassen et al. [3] found that employing user stories enhances the productivity of agile practitioners. They also reported that most practitioners who implement user stories adhere to the original template proposed by Cohn in [4]. The popularity, simplicity, and concise structure of user stories make them ideal candidates for promising natural language processing (NLP) analysis and research [5]. This widespread acceptance and straightforward nature of user stories motivate an appealing focus for studies and applications that support requirements engineers and agile practitioners. Given that user stories are inherently based on natural language, recent advancements in NLP have positioned them as ideal candidates for automated analysis and understanding through artificial intelligence and machine learning (ML) techniques [5]. However, the effectiveness of such models largely depends on the availability of high-quality, large-scale training data [6], making the development of structured user story datasets a pressing need.

Despite this need, the current research landscape reveals a significant scarcity of comprehensive user story datasets. These datasets are limited in availability, scattered across the literature, and often lack standardization. Raharjana et al. [7] highlighted several key challenges associated with existing user stories datasets. Among which is the limited publicly available data, making it difficult for researchers to obtain sufficient and representative samples. In addition, the high heterogeneity of the available datasets results from variations in story formats, sources, and writing styles, although many studies recommend Cohen's user story template [4]. Also, the manual labeling of the available datasets hinders scalability and consistent reuse. Such challenges create a difficulty in applying ML techniques (e.g. classification or semantic similarity analysis) due to the limitations mentioned earlier.

Therefore, creating a well-structured and annotated user story dataset is crucial for accelerating research on developing intelligent tools that can understand and analyze requirements in agile environments. In an effort to bridge this gap and motivated by one of the most widely used user stories datasets [8], although small, this study presents a dataset of over 10,000 user stories collected from both academic and publicly available sources through a carefully designed and systematic process. Unlike existing datasets such as Dalpiaz's [8], which are limited in diversity and size (1,680 samples) and lack NFR annotations, our dataset provides over 10,000 stories with ISO/IEC 25010-based labeling, ensuring broader domain coverage and improved utility for AI-driven requirements engineering (RE) research. This dataset was previously used in a separate research project [9] aimed at automating the identification of non-functional requirements (NFRs) in agile user stories using pre-trained language models (PLMs), and has been annotated accordingly.

Although the dataset was initially developed for NFR classification in two ongoing projects [9], [10], it offers broader value to the research and practitioner community, such as automated acceptance criteria generation or defect detection. It can support various agile RE tasks beyond NFR identification, including requirement specification, quality analysis, decision support, and enhancement of documentation and development tools. This paper enhances the credibility and utility of the dataset by providing a detailed description of its construction and annotation process. Moreover, it offers a comprehensive taxonomy of potential research and practical applications, thereby opening promising avenues for further exploration,

particularly in integrating artificial intelligence technologies with agile RE.

The remainder of the paper is organized as follows: Section II presents the background and related literature. Section III describes the dataset construction process, including data sources, annotation, and statistics. Section IV presents a taxonomy of research and practical use cases. Section V discusses key issues and considerations. Section VI presents the results and discussion, highlighting dataset characteristics, validation, and demonstrated utility in prior experiments. Section VII explores potential threats to validity. Finally, Section VIII concludes the paper.

II. Background and Relevant Literature

This section provides foundational background and highlights key works that support the need for structured user story datasets. Agile RE and user stories have been the focus of extensive research, especially with the rise of automation and AI tools. This section presents an overview of relevant background and key studies that inform the need for structured user story datasets. The background is divided into three areas: Agile RE and user story foundations, the application of NLP and large language models (LLMs), and an overview of existing datasets.

A. Agile RE and User Stories

Agile RE is a subfield of software engineering that focuses on eliciting, documenting, analyzing, and managing system requirements within Agile development environments such as Scrum and Extreme Programming (XP). These environments emphasize flexibility, short iterative cycles, and continuous stakeholder interaction, which necessitate lightweight and efficient techniques for capturing requirements that align with the fast-paced nature of Agile processes. One of the most widely used techniques in Agile RE is the user story [2], [11], which is a short textual representation describing a feature or functionality desired by the end user. User stories are typically written in a semi-structured natural language in a simple template as follows: "As a [*type of user*], I want [*goal*] so that [*benefit*]". For example: "As a customer, I want to transfer funds between my accounts, so that I can pay off my credit card."

User stories act as a communication bridge between development teams and stakeholders, serving as basic planning and estimation units in Agile sprints. However, their use faces challenges. Key issues include ambiguity in expression due to reliance on natural language, incompleteness or imprecision in capturing requirements [12], and difficulty representing non-functional requirements (e.g. security or performance) using such a simple textual format [13]. These challenges have motivated researchers to develop tools and techniques, including artificial intelligence and NLP approaches, recently, to improve the formulation, analysis, and understanding of user stories more efficiently.

B. NLP and LLM for User Stories

Raharjana et al. [7] conducted a systematic review of studies applying NLP techniques to Agile user stories. Their study extracted the NLP tools and techniques used across the reviewed literature and proposed a conceptual and thematic classification of research goals, which included discovering defects, generating software artifacts, identifying key abstractions of user stories, and tracing links between models and stories. The review also highlighted several challenges in the literature and the need to improve the performance of the proposed solutions. In addition, there is a limited availability of datasets and an ongoing reliance on manual data labeling processes. Additional issues relate to the domain dependency of models and their limited generalizability across contexts, the complexity of correctly interpreting requirement sentences, and the persistent need for human intervention in decision-making.

Amna and Poels [14] provide a comprehensive systematic mapping of academic research on the user story techniques within Agile Software Development. The authors analyzed 186 studies published between 2001 and 2021, classifying them across five dimensions: RE activity, problem type, research outcome, research type, and publication venue. Their findings reveal a strong emphasis on system design challenges, while ambiguity and collaboration received comparatively less attention. Four major research gaps were identified, including the lack of validation and evaluation of proposed solutions and the limited exploration of cognitive and interactional aspects. This study serves as a useful reference for future research on user stories, excellently allowing researchers to clearly position their contributions.

Recently, several emerging studies showed an evolving direction of exploring the integration of LLMs and Generative AI into RE. For example, Hemmat et al. [15] recognized a steadily growing interest and significant progress in leveraging LLMs for various software engineering tasks, particularly in requirement engineering. The analysis of Arora et al. [16] showed that LLMs have the potential to enhance several RE tasks by automating, streamlining, and augmenting human capabilities because of their ability to simulate stakeholder perspectives, generate alternative requirements, address requirements quality, cross-reference with standards, and generate structured documentation. Broadly, there is increasing observation that Generative AI and LLMs are reshaping the landscape of software development in general and RE in particular. We refer to Hemmat et al. [15] and Arora et al. [16] as important references for researchers and practitioners working at the intersection of RE, including user stories, and AI. These studies help streamline and clarify the current research directions in this space and briefly highlight key considerations and challenges, especially when exploring generative AI or LLMs applied to RE tasks. Also, they primarily provide directional insights that researchers need to consider in these emerging fields.

C. Agile User Stories Datasets

In the context of Agile user story research, Dalpiaz [8] released a dataset comprising 22 backlogs with a total of 1,680 user stories covering various projects and domains. This dataset has become a fundamental resource in many studies that use NLP techniques to tackle different challenges related to user stories, such as improving how they are formulated and written [5], identifying requirements like privacy [17], [18], extracting quality attributes that could influence early architectural design decisions [19], classifying NFRs using PLMs [9], and utilizing LLMs such as LLaMA and GPT-3.5 to simulate different roles while generating acceptance criteria for requirements expressed as user stories [20].

While this dataset has served as a cornerstone in many NLP and AI-driven research efforts in Agile RE, it remains limited in scope and size. There is a pressing need for larger, more diverse, and comprehensive datasets to further advance the field. Therefore, this promising direction is encouraged to be explored, as building and sharing richer datasets can significantly accelerate data-driven RE research on user stories and improve the effectiveness of Agile requirements practices.

III. Dataset Construction Process

This section outlines the construction of the aggregated dataset, beginning with the motivation behind its creation, followed by the strategy employed to gather and process the data. The complete process, including dataset sources, labeling, and preprocessing, is summarized in Fig. 1. Section IV presents a conceptual taxonomy that highlights how researchers and practitioners may utilize this dataset across various Agile RE use cases, especially in contexts involving LLMs and generative language models.

A. Motivation for Building the Dataset

As part of previous research [9], the effectiveness of fine-tuning pre-trained language models to identify non-functional requirements (NFRs) in user stories was examined. The goal was to address a key challenge in agile RE: the frequent neglect or implicit handling of NFRs and the general inability of user stories to explicitly capture them. This led us to ask: Can user stories implicitly contain NFRs even if they are not explicitly stated? Through the investigation, it was found that some user stories do, in fact, embed implicit NFRs, a finding also noted in earlier work by other authors [19]. However, that work was limited in scope due to the small and constrained dataset used, which restricted the automation potential and performance of the applied models. This highlighted the need for a larger and more diverse dataset to support further research in this area, particularly given that machine learning models typically require substantial data to perform effectively and generalize reliably in tasks such as NFR identification. This realization motivated us initially to construct a broader dataset of user stories. While the initial focus was centered on NFR detection, the dataset is now released for broader use, recognizing its potential to support a wide range of applications beyond NFR-related tasks.

B. Dataset Collection and Sources

While research on the application of artificial intelligence in Agile RE continues to grow, one persistent challenge is the scarcity of high-quality, large-scale datasets of user story requirements. Existing datasets are often limited in size, privately held, or lack sufficient diversity [7], making them inadequate for training and evaluating modern machine learning models, or even for supporting various automation and data-driven use cases. To tackle this challenge, a step was taken to build a larger dataset by gathering user stories from various academic and public sources. These include datasets from prior research, software project repositories, and curated online content formatted in user story style. Through a literature review in another project [9] and with the guidance of another review [7], the papers that used datasets in their work were identified. The datasets were then collected, and a snowball search method was used to find related works and additional relevant papers until no more could be found. Additionally, academic literature and popular data-hosting platforms like Zenodo, GitHub, and Kaggle were searched.

Table I lists all sources used in the dataset construction, along with the number of user stories contributed by each. After removing duplicates, the final dataset comprises a total of 10,006 unique user stories. This aggregated dataset is now publicly available via a GitHub repository[1].

C. Dataset Labeling

Although the primary purpose of labeling this dataset was to support another research project on identifying NFRs in agile user stories, the labeling process was designed to be methodical and general enough to serve broader research and practical applications. Given the diversity and occasionally ambiguous nature of the collected backlogs and their respective domains, a multi-stage labeling strategy was employed, illustrated in Fig. 1.

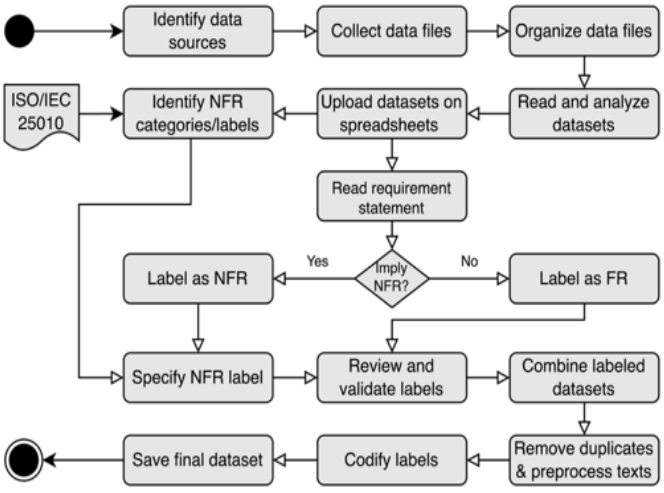


Fig. 1. Dataset construction and labeling workflow.

In the first stage, each backlog was uploaded into a spreadsheet, and a preliminary review was conducted to determine its domain and contextual background. This step was essential for understanding the scope and purpose of each

[1] https://github.com/a-re22/Agile_Req_NFRs.

backlog and for assessing whether user stories implicitly or explicitly conveyed an NFR. Once such indications were observed, the NFR labeling process was initiated. To guide the classification, the ISO/IEC 25010:2011 quality model [21], which has since been replaced by the updated version, ISO/IEC 25010:2023 [22] was used. It provides a well-structured taxonomy of software quality characteristics. It defines eight high-level NFR categories and 32 subcategories, as summarized in Table II, which offers a unified reference for annotation. Using this model, two researchers collaboratively labeled each requirement as either a functional requirement (FR) or an NFR by asking: "Does this statement explicitly or implicitly indicate an NFR?" If so, it was tagged accordingly. Each NFR-labeled story was then assigned to one of the eight NFR categories. The resulting labels were encoded in spreadsheets for consistency and clarity. The Functional Suitability category was excluded from labeling, although it is part of ISO/IEC 25010, because it was not feasible to reliably evaluate attributes related to Functional Suitability, like correctness, completeness, or appropriateness. Accordingly, FR in the dataset refers to general functional requirements rather than the functional suitability dimension.

TABLE I. SOURCES OF USER STORY SAMPLES IN THE DATASET

Data Source	Title	Description	# of samples
Dalpiaz [8]	Requirements data sets (user stories).	A collection of 22 data sets of 50+ requirements each, expressed as user stories. These were all found online or retrieved from software companies with permission to disclose. The data sets have been originally used to conduct experiments about ambiguity detection and developing Requirement Engineer Validation & Verification (REVV) tool.	1680
Murukannaiah et. al [23]	Acquiring Creative Requirements from the Crowd: Understanding the Influences of Personality and Creative Potential in Crowd RE.	This data was collected as part of research on understanding the influence of personality and creativity potential in Crowd RE. Despite the goal and other aspects in the dataset, more than 3000 SW requirements formulated as user story were extracted.	3267
Köse and Aydimer [24]	A User Story Dataset for Library and Restaurant Management Systems.	A dataset used to extract key concepts from user stories and build a knowledge graph by connecting related terms. SCOUT: Supporting User Story Completeness via Knowledge Graphs.	773
Rohmann and Paech [25]	Dataset and Application of Algorithms of 'Towards a More Set of Acceptance Criteria'.	A dataset contains SW requirements for CoronaWarn App used for experimenting a proposed algorithm aiming to a more complete set of acceptance criteria.	32
Lucassen et al. [26]	Extracting conceptual models from user stories with Visual Narrator	A dataset used to experiment an NLP-based tool (Visual Narrator) to automate extracting conceptual models from user stories.	465
Lucassen et al. [27]	Improving agile requirements: the Quality User Story framework and tool	A dataset used to evaluate Automatic Quality User Story Artisan (AQUSA) software tool- an NLP-based tool- in detecting writing quality defects within agile requirements and suggest possible remedies.	187
Prabu et al. [28]	User story-based automatic test case classification and prioritization using NLP-based deep learning	A dataset used to investigate extracting key information from user stories to facilitate automatic classification and prioritization of the associated test cases by considering factors like criticality, complexity, and dependencies.	602
Dalpiaz et al. [29]	On deriving conceptual models from user requirements: An empirical study	A dataset used to evaluate the derivation of structural conceptual models that represent the domain of the system from agile user requirements.	540
The Trident Project[2]	On the Trident Project: Architecture	A publicly available set of user stories created by a Duke University team for the Trident project which is a software to support metadata creation for their library system.	49
Online Contents	Zenodo[3], Github[4], Kaggle[5]	Searched these data hosting platforms using terms "user stories" or "agile requirements" and retrieved the publicly available datasets.	733
An ongoing project[6]	Generating requirements acceptance criteria using LLM agents and prompts engineering.	A dedicated project on using LLM agents and prompt engineering to formulate acceptance criteria and concise user stories for different roles and software domains.	1779
Duplicates	A total of 101 duplicate user stories were detected, primarily due to overlapping content across some backlogs shared between the dataset published in [8], [26], [27].		-101
Total			10,006

To ensure the reliability of the labeling process, inter-annotator agreement between the initial researcher consensus and the expert-reviewed labels was retrospectively measured using Cohen's Kappa [30]. The resulting score of 0.8335 indicates substantial agreement, affirming the consistency and validity of the annotation. Any expert-identified errors were corrected, though such cases were limited. After finalizing all labels, the datasets were merged into a unified file, duplicates were removed, and labels were converted to numeric identifiers to support downstream machine learning tasks. While this labeling effort was initially driven by a specific research focus on NFR identification, the annotated dataset is released to the broader community, recognizing its potential to support a wide variety of use cases beyond its original scope. Accordingly, each user story is stored in a separate column from its label, allowing researchers to disregard or redefine the annotations according to their own use cases. The existing labels reflect another research intent but are not binding.

[2] http://blogs.library.duke.edu/digital-collections/2009/02/13/on-the-trident-project-part-1-architecture/

[3] https://zenodo.org

[4] https://github.com

[5] https://www.kaggle.com/datasets

[6] https://github.com/a-re22/Agile_Req_NFRs

D. Dataset Preprocessing

To ensure consistency while preserving the natural linguistic structure of the user stories, only minimal preprocessing was applied. This included converting all text to lowercase and removing punctuation marks to reduce textual noise and help models focus on meaningful patterns. Also, duplicate detection and removal were performed, identifying 101 duplicated user stories, mostly resulting from overlaps among backlogs sourced from [8], [26], [27]. After eliminating these duplicates, the dataset comprised 10,006 unique entries. No stemming, lemmatization, or semantic normalization was applied, as the aim was to preserve the textual authenticity and variety of user stories. Finally, all records were merged into a unified, clean dataset, and a consistency check was performed to ensure the integrity and readiness of the data for further analysis.

TABLE II. NFR CATEGORIES BASED ON ISO/IEC 25010

ID	Abb.	Label	Related Attributes	
0	FR	Functional	- Functionality feature	
1	CM	Compatibility	- Co-existence	- Interoperability
2	MN	Maintainability	- Modularity - Reusability - Analyzability	- Modifiability - Testability
3	SE	Security	- Confidentiality - Integrity - Accountability	- Authenticity - Non-repudiation
4	PE	Performance	- Time behavior - Capacity	- Resource utilization
5	PO	Portability	- Adaptability - Installability	- Replaceability
6	RL	Reliability	- Availability - Recoverability	- Fault tolerance - Maturity
7	US	Usability	- Operability - Learnability - User interface aesthetics	- Appropriateness recognizability - User error protection - Accessibility

E. Statistics of the Final Annotated Dataset

The final annotated dataset comprises 10,006 unique user stories, with 64.5% labeled as functional requirements (FRs) and the remaining 35.5% distributed among seven NFRs. On average, each story contains 25 words, with sentence lengths ranging from a minimum of 7 words to a maximum of 154 words. Table III presents sample stories labeled according to NFR category. To maintain clarity and consistency, the general NFR category level was used to label user stories instead of the more detailed subcategory level as defined by ISO/IEC 25010. This decision was driven by time constraints and the relatively limited number of examples per subcategory. For instance, if a story clearly pertains to the confidentiality aspect of security, it is labeled simply as Security, since the broader category inherently encompasses the subcategory. This approach strikes a balance between precision and practicality, supporting meaningful analysis while ensuring reliable annotation.

TABLE III. EXAMPLE USER STORIES ACROSS FUNCTIONAL AND NFR

Label	Example from the Dataset
FR	As an archivist, I want to view collection files, so that I can gain background information for processing a collection.
CM	As a developer, I want a Mongo integration so that I can integrate packaged data with pipelines that use Mongo.
MN	As a user, I want the app to be automatically updated so that I always have the latest features and fixes.
SE	As a system administrator, I want users to log in using two-step authentication, so that authorized users are only allowed.
PE	As IFA, we want the system response time for fans team administration and IFA administration to not exceed 1 second so that it enhances the user's experience.
PO	As a user, I want to be able to access the service from both a website and a mobile app, so that I can use it on multiple devices.
RL	As the IFA, we want the system to support at least 50,000 users simultaneously, so that it does not crash during peak times.
US	As a user, I want the app to save my preferences, like language or format settings, so that I do not have to set them each time I use the service.

IV. TAXONOMY OF DATASET USE CASES

This section proposes a conceptual and structured taxonomy outlining potential use cases for this dataset, particularly for researchers in data-driven RE (RE) and, more specifically, those focused on Agile RE. This taxonomy is not intended to be exhaustive or to cover all possible research directions. Rather, it is derived from observations during the dataset collection process and inspired by the categorization of NLP applications on user stories [7], with some extensions and refinements. The objective is to offer researchers a high-level perspective that can help them position their research contributions in relation to this dataset. Below is a brief description of each category. Fig. 2 presents a concise structural overview of the proposed research directions. Although LLMs open new opportunities across these use cases, their risks (e.g. hallucination, bias, and explainability limitations) must also be considered, as further discussed in Section V.

A. Quality of User Story Writing, Rewriting, or Refinement

This category focuses on assessing and improving the writing quality of user stories. It includes adherence to formal templates such as INVEST [31], the three Cs criteria [32], and QUS frameworks [27], addressing syntactic, semantic, and pragmatic dimensions. It also covers practices like ensuring style consistency, converting informal stories into structured templates, and enhancing clarity or completeness. These use cases enable researchers to develop tools and metrics for evaluating and refining story quality, supporting clearer communication, testability, and estimation. The dataset provides diverse examples that can be used to train or benchmark models on automatic quality assessment and guided story refinement for formulating a high-quality and precise user story.

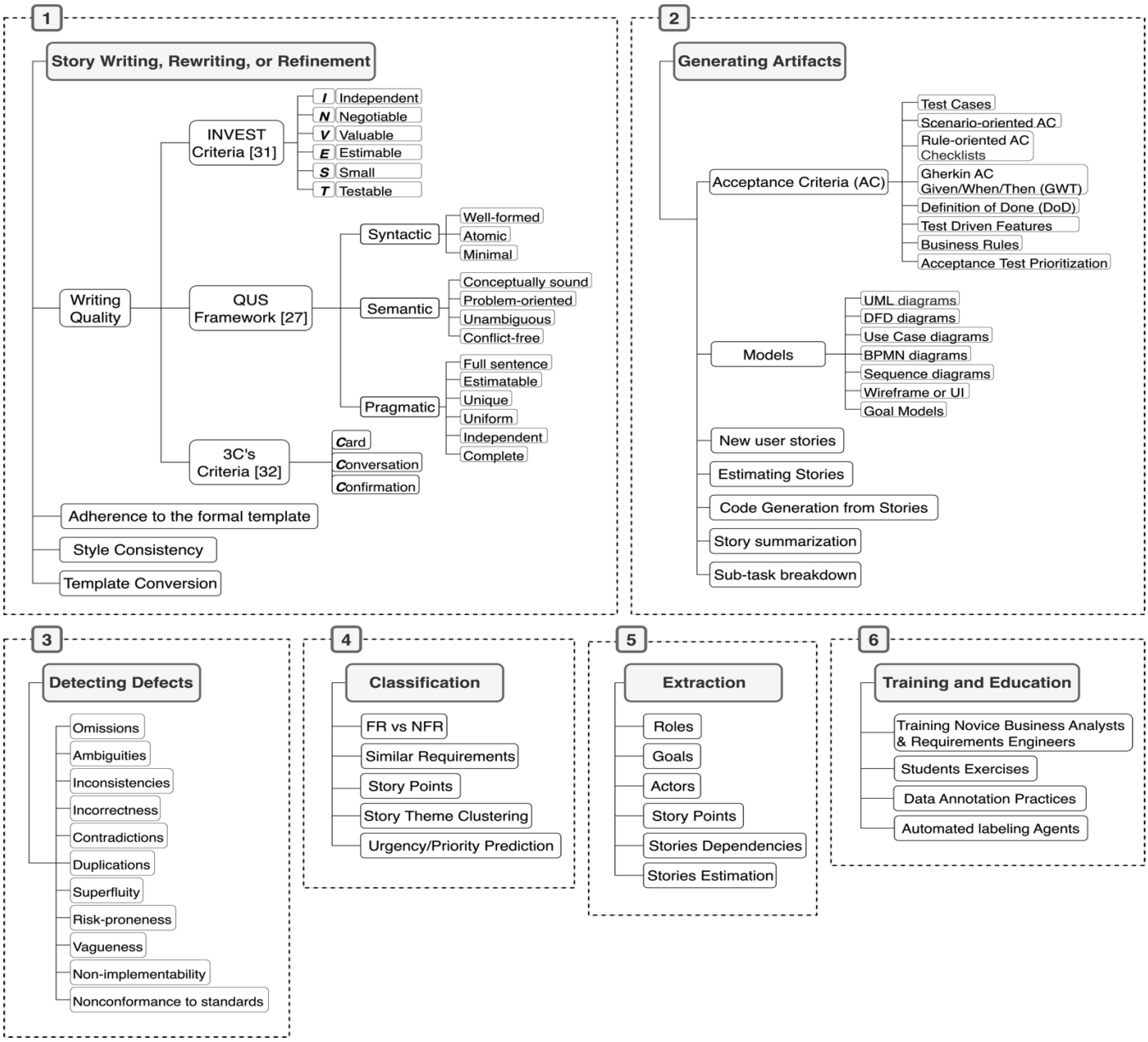


Fig. 2. Taxonomy of dataset use cases for agile requirements research.

B. Generating Artifacts from User Stories

This category covers the generation of downstream software artifacts from user stories. Various artifacts could be researched, including acceptance criteria (e.g. Gherkin [33], rule-oriented, scenario-based), test cases, UML/BPMN diagrams, and model-based artifacts like goal or sequence diagrams. For example, the dataset's labeled NFRs can train models to generate security-focused acceptance criteria (e.g. Gherkin scenarios for two-factor authentication). Other use cases include generating new user stories, decomposing stories into sub-tasks, and summarizing long stories. These transformations help automate traceability and align documentation with implementation needs. The dataset offers a strong foundation for training generative models, validating transformation pipelines, or benchmarking tools that automate requirements-to-artifact pipelines using rule-based, retrieval-based, or LLM-driven approaches.

C. Detecting and Identifying Requirements Defects

This direction targets the detection of defects in user stories that may hinder development or communication. Typical software requirements defects and their taxonomies such as [34], [35] can be very helpful in this research area. Researchers can use the dataset to develop classifiers or LLM-based prompts that detect defects such as omissions, ambiguities, inconsistencies, contradictions, and vagueness, as well as issues like non-implementability or deviation from standards, automatically. By doing so, this category supports efforts to improve the reliability and correctness of requirements documentation. The dataset offers realistic variations in quality and structure, making it suitable for training models or designing rule-based and machine learning approaches to defect detection.

D. Requirements Classification

This category involves classifying user stories based on various dimensions. Primary tasks include distinguishing between functional and NFRs, grouping similar requirements, clustering stories by theme, and predicting metadata such as story points or priority. These tasks can aid backlog management, planning, and requirement prioritization. The dataset's labeled structure supports supervised learning models for these classification tasks. Additionally, the variety in domain and writing style makes it ideal for exploring generalizability across Agile projects and evaluating classification performance of traditional ML, neural NLP, LLM, and Generative AI approaches.

E. Extraction

This category focuses on extracting structured knowledge from user stories. Targeted elements include roles, goals, actors, dependencies, and estimation features like story points. Extraction supports various downstream applications such as persona generation, automated sprint planning, or linking user stories to design models. The dataset allows for experimentation with entity recognition, role-goal modeling, and relation extraction, particularly in informal or semi-structured story formats. These tasks help bridge the gap between human-authored requirements and formal modeling or analysis tools, supporting hybrid RE environments that combine human input and AI-powered assistance.

F. Training and Education

This category emphasizes using the dataset as a resource in education and training environments. It includes applications in training novice business analysts and requirements engineers, facilitating student exercises, and practicing manual or automated annotation tasks. By engaging learners with real user stories, the dataset enables experiential learning, promotes understanding of RE concepts, and builds skills in requirement writing, evaluation, and classification. It also supports the testing and development of intelligent labeling agents. This category bridges research and pedagogy by turning real-world data into a foundation for interactive, competency-based learning in Agile RE.

V. Key Considerations for Researchers

There are several important considerations that researchers should keep in mind when working with this dataset, especially those exploring applications of LLMs and Generative AI. These considerations, summarized in Table IV, are intended to help guide the responsible and effective use of the dataset. As LLMs and generative AI are increasingly applied in Agile RE, researchers must be aware of several potential risks and challenges. Table IV summarizes these key considerations to ensure a responsible and effective application. Acknowledging and addressing these concerns is critical to maintain the quality, safety, and trustworthiness of AI-assisted tools in Agile environments.

When applying the emerging modern artificial intelligence technologies, such as LLMs and generative AI, to Agile user story datasets, researchers must account for several critical risks. These include hallucinations of facts, as LLMs can produce false responses [36], loss of context, domain misalignment, and ethical concerns such as bias or stakeholder exclusion. Additionally, generated artifacts may suffer from version drift, lack traceability, or violate privacy norms. Without proper oversight, such issues can degrade requirements quality, introduce safety risks, or mislead practitioners. Therefore, researchers must adopt mitigation strategies, such as expert validation, contextual adaptation, and anonymization, to ensure responsible and effective use of AI-driven solutions in Agile RE.

VI. Results and Discussion

A. Dataset Characteristics and Validation

The constructed dataset comprises 10,006 unique user stories, of which 6,457 (65%) are Functional Requirements (FRs) and 3,549 (35%) are Non-Functional Requirements (NFRs) as shown in Fig. 3. Within NFRs, the distribution varies substantially across categories: Usability (780) and Security (701) are the most frequent, while Maintainability (379) and Portability (381) are the least represented.

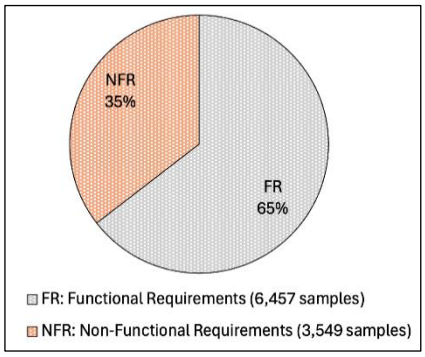


Fig. 3. Dataset FRs vs NFRs distribution.

Fig. 4 presents the Breakdown of NFR categories in the dataset. This distribution reflects common industrial practices where some NFRs, such as security, are realized more frequently than others. To ensure labeling reliability when labeling the dataset, inter-annotator agreement was measured retrospectively using Cohen's Kappa (Cohen's Kappa = 0.8335), indicating substantial agreement and validating the consistency of the classification process. This strengthens confidence in using the dataset for downstream tasks such as classification and artifact generation.

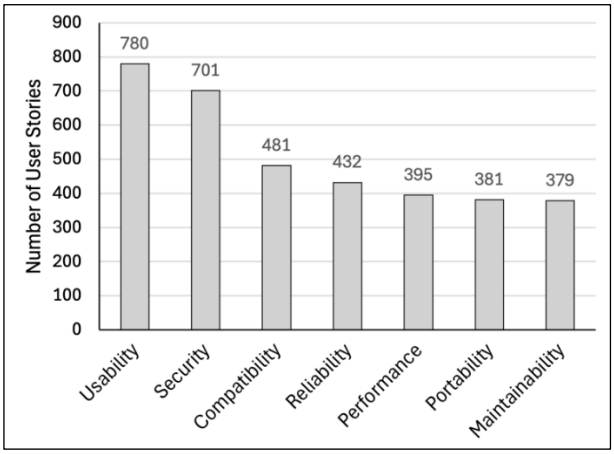


Fig. 4. NFRs distribution.

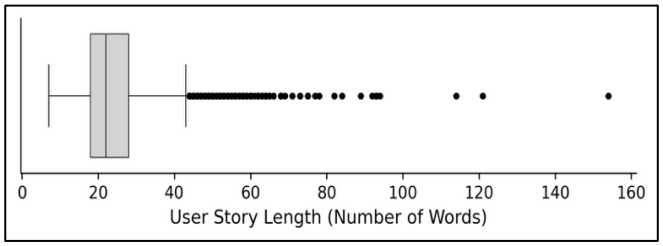


Fig. 5. Boxplot of user story lengths measured in words.

The length of user stories also shows natural variation. Fig. 5 presents a plot of user story lengths measured in words, showing an average of 23.5 words, a median of 22 words, and several long outliers reaching up to 154 words. This highlights the variability in expression style across sources. The annotated boxplot reveals several outliers, which are important to retain in order to preserve the natural linguistic diversity of agile requirements. These descriptive statistics establish the dataset's realism and its potential to support robust empirical research.

B. Demonstrated Utility in Prior Experiments

The dataset was previously employed to fine-tune transformer-based models for the automated classification of NFRs in agile user stories [9]. In these experiments, the fine-tuned BERT model achieved an F1 score of 93.4%, outperforming traditional machine learning and rule-based baselines. Comparable models such as RoBERTa, XLNet, and DistilBERT also performed competitively, underscoring the dataset's capacity to benchmark state-of-the-art approaches. Analyses further revealed that minority categories, such as Portability and Maintainability, remain challenging, largely due to data imbalance, while major categories, such as Security and Usability, achieved higher predictive performance. These results demonstrate that the dataset not only enables high-performing models but also reveals promising research applications.

TABLE IV. KEY CONSIDERATIONS FOR RESEARCHERS

Issue	Description	Implications for RE Research & Practice
Hallucination (Fabricated Output)	LLMs may produce plausible but incorrect details not present in the story or backlog.	May misguide development or testing; introduces false or unsafe assumptions. Generated outputs should undergo validation against ground-truth labels in the dataset
Project Sensitivity / Contextual Mismatch	LLMs may lack knowledge of domain-specific or organizational context, producing generic outputs.	Output may not respect business rules or constraints; increases rework.
Safety-Critical Misjudgments	In critical systems (e.g., healthcare), incorrect AI-generated requirements may cause safety risks.	Can result in non-compliance with safety-critical standards or legal risk.
Confidentiality of Requirements	User stories may expose confidential business strategies or customer data.	Risk of exposing sensitive competitive or user data through API-based models.
Data Privacy and Compliance	Stakeholder names, roles, or preferences in user stories may violate data regulations if not anonymized.	May breach regulations like GDPR, requiring careful anonymization or local deployment.
Ethical Concerns (Bias, Fairness)	LLMs may encode or introduce bias unintentionally (e.g., gender, cultural, functional bias).	Violates fairness or diversity principles in requirement framing or persona generation.
Accountability and Traceability	LLMs do not explain their reasoning, making outputs hard to audit or trace.	Challenges in proving why an AI recommendation was followed; weakens traceability.
Lack of Domain Adaptation	Generic LLMs may fail in technical or niche domains (e.g., avionics, cybersecurity).	May degrade output quality in domain-specific user stories without fine-tuning.
Validation and Verification Burden	Generated outputs still need expert review to ensure correctness and applicability.	Burden on teams to review all outputs; may nullify efficiency gains.
Overreliance or Automation Bias	Overtrust generated stories might lead to complacency or overlooked risks.	False confidence in AI can degrade quality; critical stories may be overlooked.
Incomplete Stakeholder Involvement	Reduced interaction with stakeholders may dilute shared understanding of goals.	Undermines Agile value of collaboration with customers and stakeholders.
Version Drift / Requirements Decay	Generated artifacts may go stale if not synchronized with evolving sprints or backlog.	Misaligned outputs can delay development or cause scope misunderstanding.

C. Reflections and Opportunities

The findings highlight two complementary contributions. First, the dataset itself offers a comprehensive, high-quality, and publicly available resource that consolidates fragmented prior collections into a single benchmark. Second, prior experiments confirm its utility in advancing data-driven and LLM-supported approaches in Agile-based requirements.

At the same time, the analysis surfaces future opportunities. The imbalance across NFR categories suggests the need for targeted data augmentation and rebalancing strategies. Moreover, beyond NFR identification, the dataset can support tasks such as requirements prioritization, acceptance criteria generation, defect detection, and automated documentation pipelines. These directions are encouraged to strengthen the dataset's role in aligning agile practices with high-quality software engineering outcomes.

VII. THREATS TO VALIDITY

One potential threat to internal validity lies in the manual annotation process of user stories for NFRs. Despite employing ISO/IEC 25010 and using multiple annotators with inter-rater reliability checks (Cohen's Kappa = 0.8335), subjectivity still influences the labeling decisions, especially for implicit NFRs. Variability in domain contexts and story phrasing definitely has some influence on the labelers' interpretations during their annotation process.

A threat to external validity stems from the scope and diversity of the dataset sources. While the dataset aggregates samples from multiple academic and online repositories, it still does not fully reflect domain-specific conventions across

various industries (e.g. finance, aerospace, or healthcare). As a result, the generalizability of any AI model trained on this dataset should be assessed before deployment in such specialized contexts.

VIII. Conclusion

This paper introduces a comprehensive and publicly available dataset of over 10,000 agile user stories aggregated from academic and online sources. The dataset, systematically annotated using ISO/IEC 25010-based NFR categories, was initially designed to support research on automated NFR identification. To ensure reliability, the labeling process was validated through inter-annotator agreement using Cohen's Kappa, which yielded a substantial agreement score of 0.8335, affirming the consistency of the annotations. Beyond its original purpose, the dataset holds value for a variety of tasks in agile RE, including user story classification, artifact generation, defect detection, and educational applications. A structured taxonomy of use cases was also proposed to guide future research and practical adoption of the dataset, particularly in data-driven and AI-enhanced RE.

Looking ahead, further research is needed to explore how the dataset and its annotation process can enhance the trustworthiness and scalability of automated NFR identification and classification. Future work will also investigate applications such as requirements prioritization and the formulation of acceptance criteria for NFRs in user stories. Moreover, the dataset opens up the possibility of developing automated pipelines that integrate LLMs to support practical tasks, such as requirement specification, prioritization, traceability, and acceptance criteria generation. These directions are encouraged to strengthen the validity of the dataset and its alignment with agile practices, ultimately supporting fruitful research and high-quality software engineering outcomes.

References

[1] Wang, L. Zhao, Y. Wang, and J. Sun, "The Role of Requirements Engineering Practices in Agile Development: An Empirical Study," in Requirements Engineering, D. Zowghi and Z. Jin, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 195–209. doi: 10.1007/978-3-662-43610-3_15.

[2] E.-M. Schön, J. Thomaschewski, and M. J. Escalona, "Agile Requirements Engineering: A systematic literature review," Computer Standards & Interfaces, vol. 49, pp. 79–91, 2017, doi: 10.1016/j.csi.2016.08.011.

[3] G. Lucassen, F. Dalpiaz, J. M. Werf, and S. Brinkkemper, "The Use and Effectiveness of User Stories in Practice," in Proceedings of the 22nd International Working Conference on Requirements Engineering: Foundation for Software Quality, in REFSQ 2016, vol. 9619. Berlin, Heidelberg: Springer-Verlag, 2016, pp. 205–222. doi: 10.1007/978-3-319-30282-9_14.

[4] M. Cohn, User Stories Applied: For Agile Software Development. USA: Addison Wesley Longman Publishing Co., Inc., 2004.

[5] G. Lucassen, F. Dalpiaz, J. M. E. Van Der Werf, and S. Brinkkemper, "Forging high-quality user stories: towards a discipline for agile requirements," presented at the 2015 IEEE 23rd international requirements engineering conference (RE), IEEE, 2015, pp. 126–135. doi: 10.1109/RE.2015.7320415.

[6] S. Mohammed et al., "The effects of data quality on machine learning performance on tabular data," Information Systems, vol. 132, p. 102549, 2025, doi: https://doi.org/10.1016/j.is.2025.102549.

[7] I. K. Raharjana, D. Siahaan, and C. Fatichah, "User Stories and Natural Language Processing: A Systematic Literature Review," IEEE Access, vol. 9, pp. 53811–53826, 2021, doi: 10.1109/ACCESS.2021.3070606.

[8] F. Dalpiaz, "Requirements data sets (user stories)," Mendeley Data, v1, 2018, doi: 10.17632/7zbk8zsd8y.1.

[9] A. Alhaizaey and M. Al-Mashari, "Automated Classification and Identification of Non-Functional Requirements in Agile-Based Requirements Using Pre-Trained Language Models," IEEE Access, vol. 13, pp. 87401–87417, 2025, doi: 10.1109/ACCESS.2025.3570359.

[10] A. Alhaizaey and M. Al-Mashari, "A Framework for Reviewing and Improving Non-Functional Requirements in Agile-based Requirements," in 2023 18th Iberian Conference on Information Systems and Technologies (CISTI), June 2023, pp. 1–7. doi: 10.23919/CISTI58278.2023.10211956.

[11] M. Kassab, "The changing landscape of requirements engineering practices over the past decade," in 2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE), 2015, pp. 1–8. doi: 10.1109/EmpiRE.2015.7431299.

[12] E. M. Schön, D. Winter, M. J. Escalona, and J. Thomaschewski, "Key Challenges in Agile Requirements Engineering," in Agile Processes in Software Engineering and Extreme Programming, H. Baumeister, H. Lichter, and M. Riebisch, Eds., Springer, 2017, pp. 37–51. doi: 10.1007/978-3-319-57633-6_3.

[13] I. Inayat, S. S. Salim, S. Marczak, M. Daneva, and S. Shamshirband, "A systematic literature review on agile requirements engineering practices and challenges," Computers in Human Behavior, vol. 51, pp. 915–929, 2015, doi: 10.1016/j.chb.2014.10.046.

[14] A. R. Amna and G. Poels, "Systematic Literature Mapping of User Story Research," IEEE Access, vol. 10, pp. 51723–51746, 2022, doi: 10.1109/ACCESS.2022.3173745.

[15] A. Hemmat, M. Sharbaf, S. Kolahdouz-Rahimi, K. Lano, and S. Y. Tehrani, "Research directions for using LLM in software requirement engineering: a systematic review," Frontiers in Computer Science, vol. Volume 7-2025, 2025, doi: 10.3389/fcomp.2025.1519437.

[16] C. Arora, J. Grundy, and M. Abdelrazek, "Advancing Requirements Engineering Through Generative AI: Assessing the Role of LLMs," in Generative AI for Effective Software Development, A. Nguyen-Duc, P. Abrahamsson, and F. Khomh, Eds., Cham: Springer Nature Switzerland, 2024, pp. 129–148. doi: 10.1007/978-3-031-55642-5_6.

[17] F. Casillo, V. Deufemia, and C. Gravino, "Detecting privacy requirements from User Stories with NLP transfer learning models," Information and Software Technology, vol. 146, p. 106853, 2022, doi: 10.1016/j.infsof.2022.106853.

[18] G. B. Herwanto, G. Quirchmayr, and A. M. Tjoa, "Leveraging NLP Techniques for Privacy Requirements Engineering in User Stories," IEEE Access, vol. 12, pp. 22167–22189, 2024, doi: 10.1109/ACCESS.2024.3364533.

[19] F. Gilson, M. Galster, and F. Georis, "Extracting Quality Attributes from User Stories for Early Architecture Decision Making," in 2019 IEEE International Conference on Software Architecture Companion (ICSA-C), 2019, pp. 129–136. doi: 10.1109/ICSA-C.2019.00031.

[20] Y. Li, J. Keung, Z. Yang, X. Ma, J. Zhang, and S. Liu, "SimAC: simulating agile collaboration to generate acceptance criteria in user story elaboration," Automated Software Engineering, vol. 31, no. 2, p. 55, June 2024, doi: 10.1007/s10515-024-00448-7.

[21] ISO/IEC 25010:2011, "Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models," International Organization for Standardization, Standard. [Online]. Available: iso.org

[22] International Organization for Standardization, ISO/IEC 25010:2023 — Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — Product Quality Model, Geneva, Switzerland., 2023. [Online]. Available: https://www.iso.org/standard/78176.html

[23] P. K. Murukannaiah, N. Ajmeri, and M. P. Singh, "Acquiring Creative Requirements from the Crowd: Understanding the Influences of Personality and Creative Potential in Crowd RE," in 2016 IEEE 24th International Requirements Engineering Conference (RE), Beijing, China: IEEE, Sept. 2016, pp. 176–185. doi: 10.1109/RE.2016.68.

[24] S. G. Köse and F. B. Aydemir, "A User Story Dataset for Library and Restaurant Management Systems." Zenodo, Jan. 12, 2023. doi: 10.5281/ZENODO.7529130.

[25] A. Rohmann and B. Paech, "Dataset and Application of Algorithms of 'Towards a More Set of Acceptance Criteria.'" Zenodo, Nov. 24, 2022. doi: 10.5281/ZENODO.7358697.

[26] G. Lucassen, M. Robeer, F. Dalpiaz, J. M. E. M. Van Der Werf, and S. Brinkkemper, "Extracting conceptual models from user stories with Visual Narrator," Requirements Eng, vol. 22, no. 3, pp. 339–358, Sept. 2017, doi: 10.1007/s00766-017-0270-1.

[27] G. Lucassen, F. Dalpiaz, J. M. E. M. Van Der Werf, and S. Brinkkemper, "Improving agile requirements: the Quality User Story framework and tool," Requirements Engineering, vol. 21, no. 3, pp. 383–403, 2016, doi: 10.1007/s00766-016-0250-x.

[28] A. Prabu, L. A. Suruthi, B. Sabarish, and K. Arun, "User story-based automatic test case classification and prioritization using natural language processing-based deep learning," IEEE Potentials, pp. 2–11, 2024, doi: 10.1109/MPOT.2023.3342366.

[29] F. Dalpiaz, P. Gieske, and A. Sturm, "On deriving conceptual models from user requirements: An empirical study," Information and Software Technology, vol. 131, p. 106484, Mar. 2021, doi: 10.1016/j.infsof.2020.106484.

[30] J. Cohen, "A coefficient of agreement for nominal scales," Educational and psychological measurement, vol. 20, no. 1, pp. 37–46, 1960.

[31] B. Wake, "INVEST in good stories, and SMART tasks," XP123. Accessed: Apr. 04, 2025. [Online]. Available: https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/

[32] R. Jeffries, "Essential XP: Card, Conversation, Confirmation." Accessed: Apr. 04, 2025. [Online]. Available: https://ronjeffries.com/xprog/articles/expcardconversationconfirmation/

[33] M. Wynne, A. Hellesoy, and S. Tooke, The cucumber book: behaviour-driven development for testers and developers. The Pragmatic Programmers LLC, 2017.

[34] F. Shull, I. Rus, and V. Basili, "How perspective-based reading can improve requirements inspections," Computer, vol. 33, no. 7, pp. 73–79, 2000, doi: 10.1109/2.869376.

[35] A. A. Alshazly, A. M. Elfatatry, and M. S. Abougabal, "Detecting defects in software requirements specification," Alexandria Engineering Journal, vol. 53, no. 3, pp. 513–527, 2014, doi: 10.1016/j.aej.2014.06.001.

[36] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, Y.-Y. Liu, and L. Yuan, "LLM lies: Hallucinations are not bugs, but features as adversarial examples," arXiv preprint arXiv:2310.01469, 2023.

Comparative Evaluation of Centrality Measures for Detecting Significant Nodes in Social Networks

Hardeep Singh*, Supreet Kaur, Karman Singh Sethi, Jai Sharma
Department of Computer Engineering and Technology, Guru Nanak Dev University, Amritsar, India

***Abstract*—Social networks are certainly a crucial platform for bringing people together globally. Detecting the significant nodes inside the social network remains an open problem because of the broad variety of network sizes. To solve this problem, different centrality measures have been introduced. Detecting the significant nodes is essential for speeding up or slowing down the spread of information, managing diseases and rumors, and more. This paper presents a comparative evaluation of 12 centrality measures to determine the most effective measure on the basis of accuracy, differentiation capability, and runtime. To validate performance, a series of experiments is conducted on four social networks using the validation metrics such as monotonicity, the SIR model, and Kendall tau. The experimental outcomes indicate that the gravity-based measures have superior accuracy and differentiation capability as compared to other measures. Finally, this paper outlines future research directions for enhancements based on centrality measures.**

Keywords*—Social networks; centrality measures; significant nodes; validation metrics; extended gravity

I. Introduction

Social networks depict the structure of actual social systems in which vertices denote individuals and edges denote their connections. Globally, billions of people own smartphones, and they are adept at using social media applications, including Facebook, Threads, Bluesky, and others, to communicate and disseminate information. Consequently, social networks offer an environment for promoting items, disseminating information, and other purposes. Table I lists a few instances of social networks. Many tools for exploring social networks have been provided by technology in recent years, like NetworkX, Pajek, VOSviewer, etc.

Detecting the significant nodes is crucial for carrying out the aforementioned tasks, like promoting products, spreading information, etc., as a node with a better spreading capacity is regarded as significant in social networks [1]. Significant nodes have a larger effect on the topology of the network in contrast to the remaining nodes [2]. Hence, detecting the significant nodes aids in a thorough analysis of the whole network. For instance, in the management of disease spread, we can control the spread of disease by quickly finding the most critical node of disease transmission [3]. The significant node inside a network is designated by its centrality [4]. As a result, determining the measure of centrality is crucial. Different centralities have been stated due to the enormous growth of social networks, like closeness [5], degree [6], betweenness [7], k-shell decomposition [8], gravity [9], etc. These measures can be classified as local, semi-local, and global, based on node connectivity [10], as displayed in Fig. 1.

Local knowledge regarding the node is utilized to develop local centralities. For instance, degree [6], ProfitLeader [11], etc. Recent years have seen the development of a novel class of centralities known as semi-local, which can retain a higher accuracy than local centralities. For instance, h-index [12], semi-local centrality [13], local gravity model [14], etc. Centralities that necessitate the whole framework of a network are known as global. For instance, eigenvector [6], PageRank [15], global and local structure [16], electric potential [17], etc. This paper evaluates 12 centralities and highlights the most effective measure. The findings suggest that the gravity-based measures, that is, gravity centrality and extended gravity centrality, have superior accuracy and differentiation capability as compared to other measures that include eigenvector, degree, closeness, betweenness, PageRank, h-index, k-shell, ProfitLeader, global and local structure, and electric potential.

This paper's remaining sections are laid out this way: Section II discusses the related studies, Section III describes several centrality measures, and Section IV presents the experimental methodology. Lastly, Sections V and VI conclude the paper and highlight future directions.

TABLE I. Few Instances of Social Networks

Social Networks	
Examples	***Applications***
Friendship	WhatsApp, Facebook, Twitter, etc.
Interaction	Emails, Snapchat, Phone Calls, etc.
Co-authorship	DBLP, ScienceDirect, Wikibooks, etc.
Follower	LinkedIn, Twitter, etc.

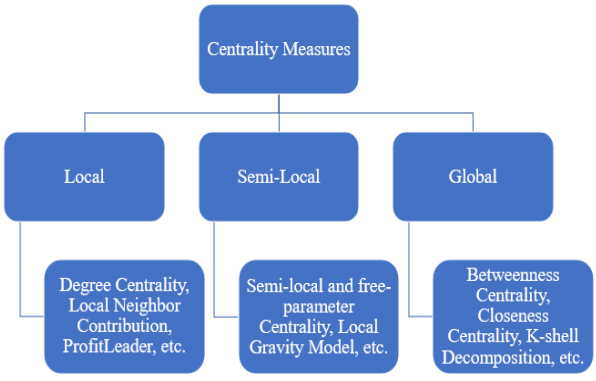


Fig. 1. Classification of centrality measures.

*Corresponding author.

II. Related Studies

In recent years, discovering the highly significant nodes in social networks has emerged as an important research challenge. Many centrality measures have been designed for detecting significant nodes, and each measure has advantages and disadvantages. Bavelas originated the concept of a centrality measure for connected networks [18]. Freeman built three well-known measures of centrality: closeness, betweenness, and degree [7]. Closeness is the global measure that considers both the direct and indirect connections. Betweenness is also the global measure that is used to identify the bridge nodes. Both these global measures have a high computational cost. The local measure, referred to as degree, considers only direct connections. A node becomes significant according to Bonacich's eigenvector measure if it has significant neighbors [6]. Search engines widely and efficiently use the global measure known as PageRank [15], which Brin & Page invented, to rank web pages. A web page attains high quality when many high-quality web pages link to it.

A semi-local measure called the h-index was created [12]; according to this measure, a node is considered more significant when it is linked to many large-degree neighboring nodes. In order to investigate human brain networks, a unique measure known as leverage centrality was invented [19]. Kitsak et al. designed the global measure called the k-shell that has low computational cost [8]. Bae & Kim invented the coreness measure, which calculates a node's influence by considering its neighbors' k-shell values [20]. Ma et al. built gravity-based measures that depend on Newton's gravitational formula [9]. A k-shell hybrid technique was created to detect the significant nodes from both the outer shells and the core [21]. Yu et al. introduced ProfitLeader, a measure that evaluates the importance of a node through its profit capacity [11]. A node's importance increases when it offers greater profit for others.

A method based on local neighbor contribution was suggested, which can be utilized with networks of different sizes and has a relatively low computational cost [22]. Sheng et al. suggested a measure called global and local structure that takes into account both local influence and global influence [16]. Through the use of the effective distance, a gravity model was developed that integrates static as well as dynamic information [23]. To detect the significant nodes, extended hybrid characteristic centrality was designed by using an extended version of the degree and E-shell technique [24]. A gravity-based model was designed that incorporates degree, k-shell, and eigenvector centrality measures [25]. An electric potential-based measure was developed that includes both local and global features of networks [17]. A method named OVED-Rank was suggested that uses orbital velocity formula and effective distance to detect significant nodes in social networks [26]. The previous studies on different centrality measures are presented in Table II.

TABLE II. Previous Studies on Different Centrality Measures

Centrality Measure	Category	Strengths	Weaknesses/ Future Work
Degree [6]	Local	Simple with low computational cost	Low accuracy
Eigenvector [6]	Global	Concentrates on both the quantity and quality of neighboring nodes	Applicable solely to connected networks
H-index [12]	Semi-local	Better node ranking than degree	Poor differentiation capability
Gravity and its enhanced variant [9]	Global	Superior accuracy	High computational cost
ProfitLeader [11]	Local	Low computational cost	Not very effective for small networks
Local gravity model [14]	Semi-local	Introduces the truncation radius R to minimize time complexity	To compute the truncation radius R as an auxiliary parameter
Global and local structure [16]	Global	Superior accuracy	High computational cost
Local and global influence [27]	Global	Superior accuracy and differentiation capability	Less efficiency
Effective distance gravity model [23]	Global	More comprehensive	High computational cost
Extended hybrid characteristic [24]	Global	Superior accuracy and differentiation capability	Restricted to undirected unweighted networks
OVED-Rank [26]	Global	Superior accuracy	Restricted to undirected unweighted networks

III. Preliminary Concepts

This section provides 12 currently and previously created centralities. A social network is represented as a graph $G = (V, E)$, where V and E designate the node set and edge set, respectively. Various measures are given as follows:

A. Degree

It represents a simple approach for calculating a node's importance by merely taking into account its closest neighbors [6]. The degree centrality C_D [7] of a node w can be computed as:

$$C_D(w) = \frac{\sum_{x=1}^{N} a_{wx}}{N-1} \quad (1)$$

where, $A = \{ a_{wx}\}$ designates the adjacency matrix, $a_{wx} = 1$ if w and x are connected, and 0 if not; N designates the node count.

B. Betweenness

The count of shortest paths that travel through a node w determines its betweenness centrality C_B [7], which is computed as:

$$C_B(w) = \frac{2}{(N-1)\,(N-2)} \sum_{i \neq w, i \neq j, w \neq j} \frac{g_{ij}(w)}{g_{ij}} \quad (2)$$

where, $g_{ij}(w)$ designates the number of shortest paths joining nodes i and j that travel through a node w.

C. Closeness

It calculates a node's closeness to every other node in the network [5]. The closeness centrality C_c [7] can be computed as:

$$C_c(w) = \frac{N-1}{\sum_{x \neq w} d(w,x)} \quad (3)$$

where, $d(w,x)$ designates the length of the shortest path between nodes w and x.

D. Eigenvector

It measures a node's importance by taking into account both the number of its adjacent nodes and their importance [6]. The eigenvector centrality C_E can be computed as:

$$Az = \lambda z, C_E(w) = z_w = \frac{1}{\lambda}\sum_{x=1}^{N} a_{wx} z_x \quad (4)$$

where, z designates the eigenvector and λ designates the constant.

E. PageRank

It is a variant of the eigenvector measure [15]. It demonstrates that a web page's importance can be measured based on both the count of linking pages and the quality of those pages. The PageRank $C_{PR}^{(k)}$ of a node w at the k-th iteration is computed as:

$$C_{PR}^{(k)}(w) = \sum_{x=1}^{N} a_{xw} \frac{C_{PR}^{(k-1)}(x)}{k_x^{out}} \quad (5)$$

where, k_x^{out} designates the count of edges from node x to w.

F. K-Shell Decomposition

It calculates the importance of a node according to its topological position [8]. It allocates each node a k-shell index through an iterative process. The most significant node holds the maximum k-shell value and is located close to the network's core. Core nodes are more significant than non-core nodes. Within the same core level, it cannot distinguish between influential nodes.

G. H-Index

The H-index (Hirsch index) was first employed to evaluate the scholarly contributions of scholars or journals according to their scholarly works and citation counts [28]. The H-index C_{HI} of a node w is expressed as the highest value h such that w has at least h neighbors for each with a degree no less than h [12]. A node's higher H-index value specifies its greater influence.

H. Gravity

It is a hybrid measure, which makes use of the path information as well as neighborhood information [9]. The gravity centrality C_G can be computed as:

$$C_G(w) = \sum_{x \epsilon \psi_w} \frac{KS(w)KS(x)}{d^2(w,x)} \quad (6)$$

where, $KS(w)$ and $KS(x)$ designate the k-shell value of nodes w and x, respectively, and ψ_w designates the set of third-order neighbors.

I. Extended Gravity

This enhanced variant of gravity measure computes a node's importance by summing the influence $C_G(x)$ of all nearest neighbors [9]. The extended gravity centrality C_{EG} can be computed as:

$$C_{EG}(w) = \sum_{x \epsilon \Lambda_w} C_G(x) \quad (7)$$

where, Λ_w designates the set of adjacent nodes of a node w.

J. ProfitLeader

ProfitLeader uses each node's profit capacity to evaluate its influence [11]. The profit capacity C_{PL} can be computed as:

$$C_{PL}(w) = \sum_{x \epsilon \Gamma_w} AR_{(w \to x)} \cdot SP_{(w \to x)} \quad (8)$$

where, Γ_w designates the neighbors of node w, $AR_{(w \to x)}$ designates the available resource of a node w for another node x, and $SP_{(w \to x)}$ designates the sharing probability (or similarity) between two nodes w and x.

K. Global and Local Structure

It computes the node's influence with two key factors: local influence and global influence [16]. The global and local structure C_{GLS} is computed as:

$$C_{GLS}(w) = I_G(w) \times I_L(w) \quad (9)$$

The global influence $I_G(w)$ can be computed as:

$$I_G(w) = D(w) \sum_{x \epsilon Nei_w} pow(A, \#Com(w,x)) \quad (10)$$

where, A is a constant, $\#Com(w,x)$ designates the count of mutual neighbors of nodes w and x, and $D(w)$ designates the degree and is computed as:

$$D(w) = \sum_{x=1}^{N} a_{wx} \quad (11)$$

The other factor is local influence. The $I_L(w)$ is computed as:

$$I_L(w) = \sum_{x \epsilon Nei_w} C_D(x) p(x) \quad (12)$$

where, Nei_w designates all nearest neighbors of a node w, and every neighbor x has a probability p of contributing to node w.

L. Electric Potential Centrality

It relies on the principle of electric potential, which represents the network's structural features on both a local and global level [17]. The electric potential C_{EP} of a node w is computed as:

$$C_{EP}(w) = LOC(w) \times GOC(w) \quad (13)$$

where, $LOC(w)$ and $GOC(w)$ designate the locality and globality of a node w, respectively.

The $LOC(w)$ is computed as:

$$LOC(w) = D(w) \times e^{\frac{D(w) + \varphi(w)}{N}} \quad (14)$$

where, e designates the base of the natural logarithm, and $\varphi\ (w)$ is computed as:

$$\varphi\ (w) = \frac{2\times\alpha}{(|D(w)-1|+1)} \quad (15)$$

where, α designates a tunable coefficient.

The $GOC(w)$ is computed as:

$$GOC(w) = \frac{D(x)\times\beta}{d(w,x)} \quad (16)$$

where, β designates a tunable coefficient.

So, the $C_{EP}(w)$ is computed as:

$$C_{EP}(w) = \sum_{x\epsilon\gamma_w}(D(w)\times e^{\frac{D(w)+\varphi\ (w)}{N}}) \times\frac{D(x)\times\beta}{d(w,x)} \quad (17)$$

IV. Experimental Methodology

This section assesses the 12 centrality measures utilizing the three parameters mentioned below:

a) Accuracy: It reflects how effectively the measure's ranking order correlates with the true ranking order [27].

b) Differentiation Capability: It depicts the effectiveness of the measure to discriminate the nodes' importance [27].

c) Efficiency: It indicates the runtime required by the centrality for evaluating the importance of the nodes [27].

A. Validation Metrics

1) SIR Model: It is applied to determine the propagating influence of ranked nodes [29], [30] and has three states:

- Susceptible *(S)*: People with good health who are prone to contamination from other people are included in this state.
- Infected *(I)*: People with the illness may be likely to infect other people.
- Recovered *(R)*: After recovering, people with the illness are unable to infect other people.

All nodes are initially within the *S* state, with just node *y* within the *I* state. Every time step, the *I* nodes aim to infect those around them within the *S* state with an infection probability β. Nodes within the *S* state move to the *I* state upon infection. The *I* nodes then move to the *R* state with a recovery probability μ. This cycle repeats until the network contains no *I* nodes. Once all *I* nodes have shifted to the *R* state, we count the total *R* nodes to quantify the influence of node *y*, as displayed in Fig. 2. The threshold value of β can be computed as:

$$\beta_{th} = \frac{<K>}{<K^2>-<K>} \quad (18)$$

where, $< K >$ designates the average degree and $< K^2 >$ designates the second-order average degree [31].

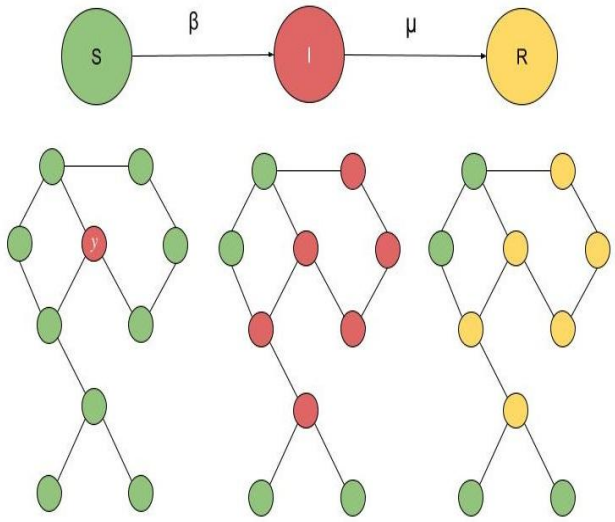


Fig. 2. The SIR model.

2) Kendall Tau τ: Kendall τ [32], [33] is applied to quantify the correlation of the ranking list of the measure with that of the SIR model. Suppose two ranking lists, Q and R, are considered for correlation and have similar nodes $N: Q = (q_1, q_2, \dots, q_N)$ and $R = (r_1, r_2, \dots, r_N)$.

Kendall τ between Q and R is computed as:

$$\tau(Q,R) = \frac{N_c-N_d}{0.5N(N-1)} \quad (19)$$

where, N_c and N_d designate the count of concordant and discordant pairs, respectively. A ranked list produced by centrality is more accurate when the τ value is higher [3].

B. Data Description

The topological properties of social networks are mentioned in Table III. The graphical visualization of four networks using the NetworkX Python Package is displayed in Fig. 3. These networks are openly accessible and available for download from the site https://networkrepository.com/

1) LastFM: In March 2020, the public API was used to gather this LastFM user social network, with the nodes signifying LastFM members from Asian nations and the edges signifying reciprocal follower ties.

2) Power: This is a representation of the structure of the US power grid in the Western States, where nodes signify power stations and edges signify lines between them.

3) Webspam: The Purdue University graph database created this dataset. Nodes are signified by web pages, and edges by hyperlinks.

4) Dolphins: It describes the network of bottlenose dolphins. Every edge signifies a dolphin-to-dolphin link, and every node signifies a bottlenose dolphin.

TABLE III. TOPOLOGICAL PROPERTIES OF FOUR SOCIAL NETWORKS

Network	N (Node count)	M (Edge count)	K_{max} (Maximum degree)	$<K>$ (Average degree)	$<CC>$ (Average clustering coefficient)	β_{th} (Infection threshold)
LastFM	7624	27806	216	7.2943	0.2194	0.0409
Power	4941	6594	19	2.6691	0.0801	0.3483
Webspam	4767	37375	477	15.6807	0.2860	0.0140
Dolphins	62	159	12	5.1290	0.2590	0.1723

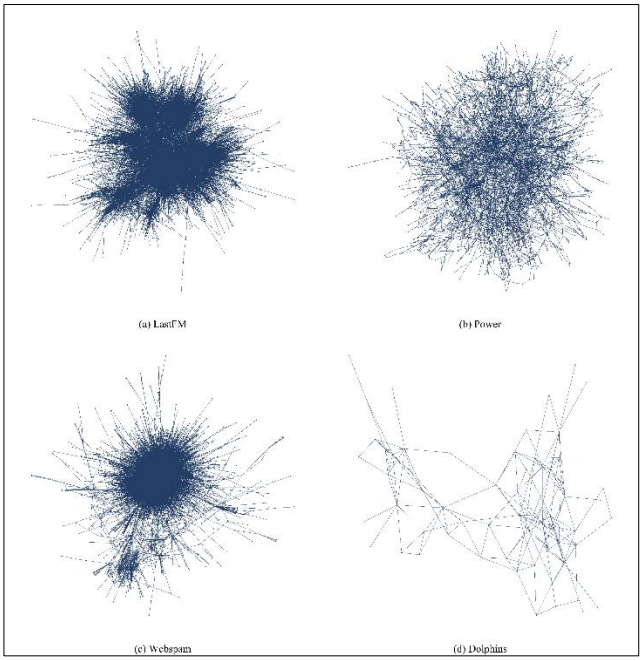


Fig. 3. Graphical visualization of four networks using the NetworkX Python Package.

C. Results

We have employed the validation metrics to compare the effectiveness of 12 measures in detecting the significant nodes. Firstly, we employ a sample network, consisting of 15 nodes and 18 edges, as displayed in Fig. 4. By utilizing this sample network, we first obtain the ranking lists based on 12 measures and the SIR model, as depicted in Table IV. We then compare the ranking list of each measure with that of the SIR model to identify the common nodes between the two lists. With this comparison, we observe that the number of common nodes for C_G, C_{EP}, C_{EG}, C_{PL}, C_C, C_{GLS}, C_{HI}, C_B, C_E, C_{PR}, C_{KS}, and C_D with the SIR model is 10, 7, 6, 6, 5, 5, 4, 3, 3, 2, 2, and 1, respectively. According to the above analysis, it can be seen that the C_G measure has the most common nodes, which means the ranking list attained from the C_G measure and the SIR model match well. Therefore, the C_G measure detects the significant nodes better than the other measures. In addition, we compare these measures on the basis of Kendall τ, as depicted in Fig. 5, and find that the C_G measure performs superior to the other measures.

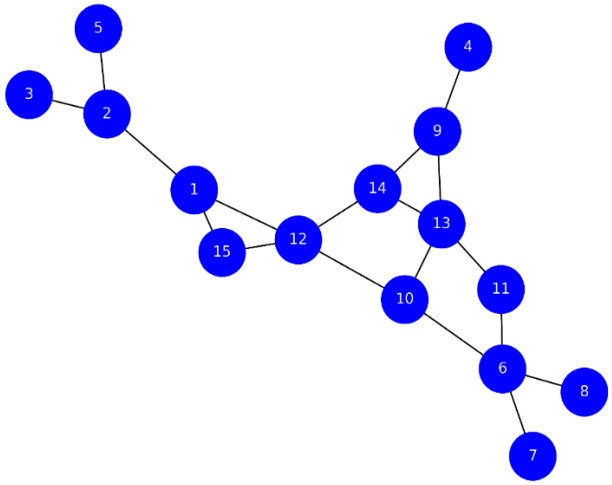


Fig. 4. A sample network with 15 nodes and 18 edges.

TABLE IV. SAMPLE NETWORK RANKING LISTS

Rank	C_D	C_B	C_C	C_E	C_{PR}	C_{HI}	C_{KS}	C_G	C_{EG}	C_{PL}	C_{GLS}	C_{EP}	SIR
1	6	12	12	13	6	13	1	12	13	12	13	12	12
2	12	1	10	12	12	14	6	13	12	13	12	13	13
3	13	10	14	14	13	10	14	10	10	6	10	6	10
4	10	6	13	10	2	12	12	14	14	14	14	10	14
5	9	2	1	9	1	1	11	6	9	1	6	14	6
6	2	13	6	6	9	6	10	1	1	10	1	1	1
7	1	14	15	1	10	9	9	9	6	9	9	9	9
8	14	9	9	11	14	15	15	11	11	2	11	2	15
9	11	11	11	15	11	11	13	15	15	11	2	11	11
10	15	4	2	2	15	7	7	2	2	15	15	15	2
11	3	3	7	4	3	3	3	7	7	3	7	7	8
12	7	7	8	7	5	2	2	8	8	4	8	8	4
13	5	5	4	8	7	5	5	4	4	5	4	4	7
14	4	8	5	5	8	4	4	5	5	7	3	5	5
15	8	15	3	3	4	8	8	3	3	8	5	3	3

The research work involves four experiments based on the three parameters outlined below:

1) Accuracy:

a) Experiment: Comparison of centrality measures using Kendall's τ values.

Kendall's τ serves as a validation metric to quantify the accuracy of centralities. The accuracy of the 12 measures, evaluated by Kendall's τ, is shown in Table V. Table V highlights that C_{EG} and C_G, the gravity-based measures, are highly effective and deliver the best outcomes among all 12 measures. The extended gravity measure performs the best, while the gravity measure ranks second. Among all measures, the C_B and C_{PR} measures perform the worst, with minimal correlation in most networks. Table V emphasizes the three best-performing measures in bold.

TABLE V. VALUES OF KENDALL TAU FOR 12 MEASURES

Network	C_D	C_B	C_C	C_E	C_{PR}	C_{HI}	C_{KS}	C_G	C_{EG}	C_{PL}	C_{GLS}	C_{EP}
LastFM	0.5829	0.4112	0.6821	0.6761	0.4222	0.6016	0.6101	**0.8081**	**0.8468**	0.5779	**0.7456**	0.6617
Power	0.4401	0.3275	0.3850	**0.6306**	0.2441	0.4188	0.3451	**0.6843**	**0.7906**	0.4327	0.6065	0.5498
Webspam	0.7124	0.5132	0.8093	**0.8456**	0.5274	0.7267	0.7315	0.8252	**0.8592**	0.7011	**0.8329**	0.7742
Dolphins	0.7472	0.5410	0.6584	0.6986	0.6647	0.7451	0.5219	**0.8488**	**0.8974**	0.7409	0.8202	**0.8308**

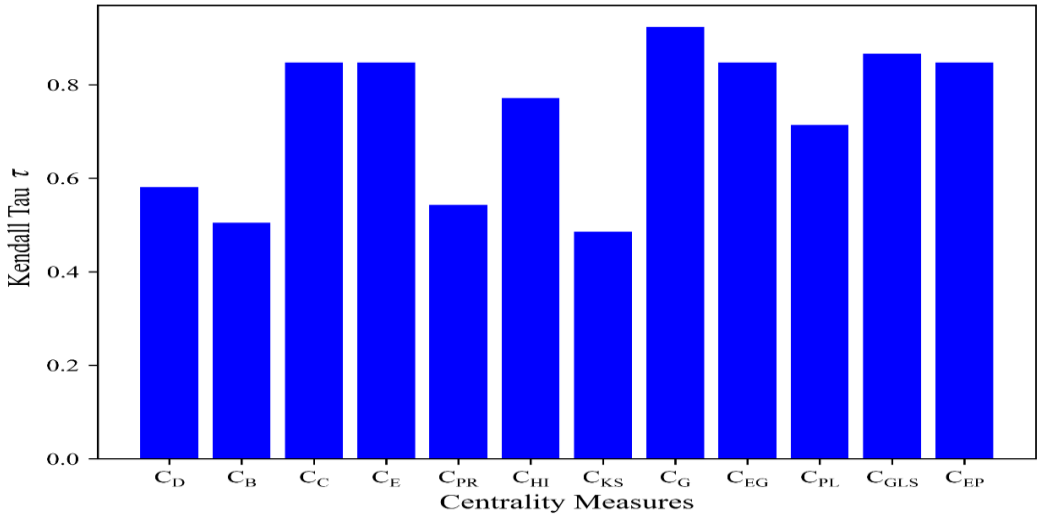


Fig. 5. Kendall τ outcomes of the sample network.

b) Experiment: Comparison of the centrality measures based on propagation influence.

To assess the effectiveness of 12 centralities, we compute the propagating ability of ranked nodes by employing the SIR model. For differentiating the significant nodes, each node's influence is first computed using the 12 centrality measures and then ranked in descending order. Second, every ranked node is utilized as an initial infected node (seed node) to infect susceptible nodes with an infection probability β. Finally, the count of nodes infected by each seed node is computed. The transmission rate of ranked nodes is plotted across the networks, as depicted in Fig. 6. The ranked index signifies the nodes' ranking order in descending order according to their influence, whereas *F(t)* signifies the total of infected and recovered nodes at time *t*. The SIR model states that highly significant nodes have the capacity to infect a greater number of other nodes. For a measure to be efficient, its infection curve should show a consistent reduction in the number of infected nodes when the node influence decreases from left to right. As shown in Fig. 6, the C_{EG} measure exhibits a remarkable infection capability compared to the other measures across all networks, with its curves appearing smooth and stable with minimal fluctuations. The C_G measure also performs well in all networks, but not as well as the C_{EG} measure. In contrast, the infection curves of the C_B and C_{PR} measures display numerous peaks and troughs in all networks. Thus, both C_B and C_{PR} measures show the worst performance.

2) Differentiation capability:

a) Experiment: Comparison of centrality measures based on monotonicity *M* values.

The monotonicity *M* [20] is applied to quantify the differentiation ability of centrality measures and is computed as:

$$M(Y) = \left[1 - \frac{\sum_{r \epsilon Y} N_r(N_r-1)}{N(N-1)}\right]^2 \quad (20)$$

where, Y designates the ranking pattern, and N_r designates the node count with the identical rank r.

The monotonicity *M* values fall between 0 and 1. The higher the monotonicity value, the better the differentiation capability. In Table VI, it can be clearly shown that the C_{EP}, C_E, C_{PR}, and C_{EG} measures have the best differentiation capability as compared to other measures. In every network, the C_{KS} and C_{HI} measures exhibit the poorest performance. Table VI emphasizes the best-performing measures in bold.

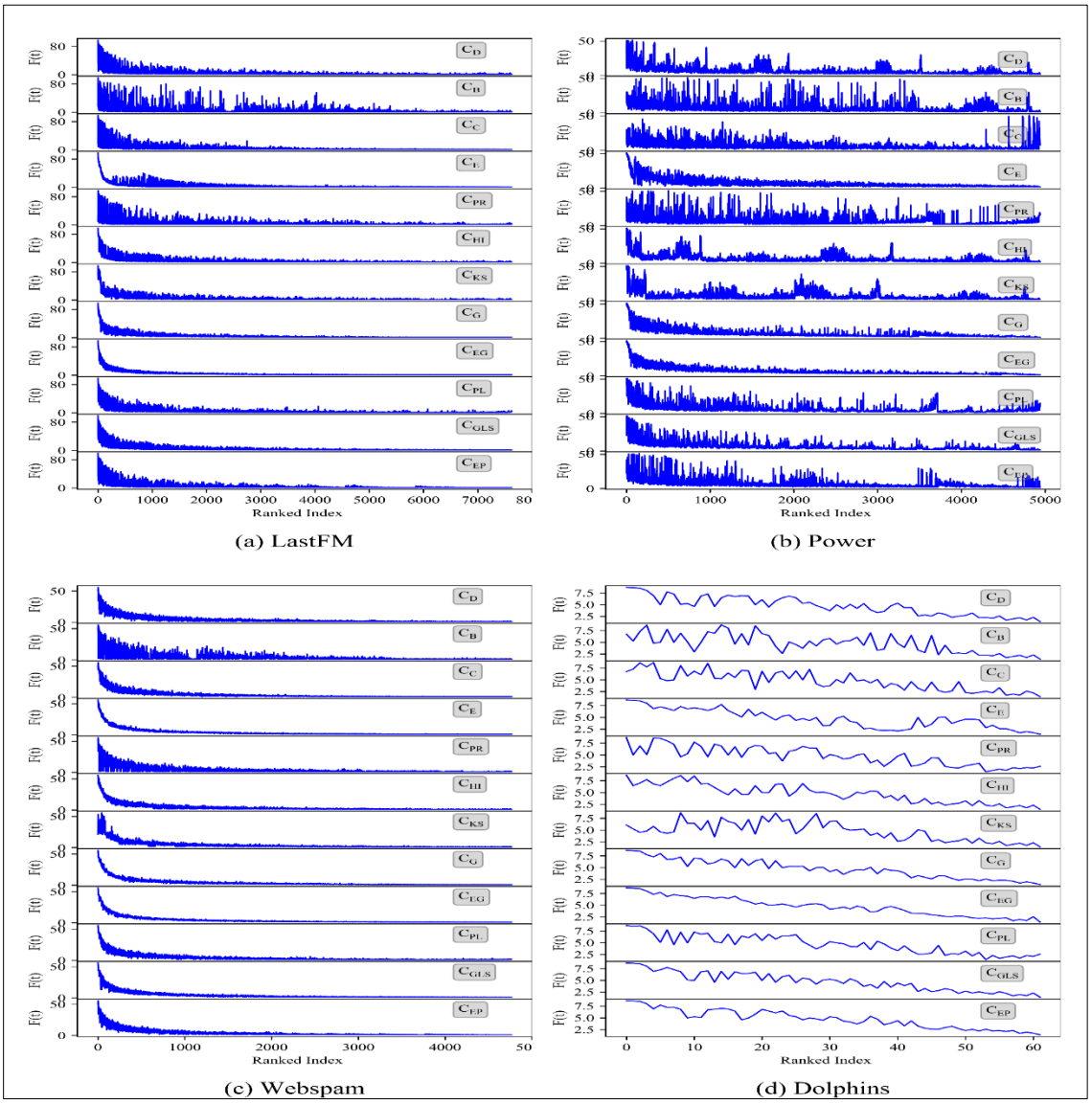


Fig. 6. Influence of propagation on the rankings using 12 measures.

TABLE VI. MONOTONICITY VALUES FOR 12 MEASURES

Network	C_D	C_B	C_C	C_E	C_{PR}	C_{HI}	C_{KS}	C_G	C_{EG}	C_{PL}	C_{GLS}	C_{EP}
LastFM	0.7978	0.8347	0.9997	**0.9999**	**0.9999**	0.7570	0.7348	0.9998	**0.9999**	0.9959	0.9975	**0.9999**
Power	0.5927	0.8319	0.9998	**0.9999**	**0.9999**	0.3930	0.2460	0.9979	**0.9997**	0.9624	0.9754	**0.9999**
Webspam	0.8640	0.8471	0.9996	**0.9998**	**0.9998**	0.8455	0.8374	0.9998	**0.9998**	0.9981	0.9988	**0.9998**
Dolphins	0.8312	0.9623	0.9737	**0.9979**	**0.9979**	0.6841	0.3769	0.9979	**0.9979**	0.9905	0.9958	**0.9979**

3) Efficiency

a) Experiment: Comparison of centrality measures using runtime.

In this experiment, we have analyzed the runtime of 12 measures in four networks to measure the efficiency of each centrality measure. As shown in Table VII, the run time of the C_D measure is significantly lower than that of the other measures across all networks. The C_{KS} and C_{HI} measures also have a low runtime after the C_D measure in all networks. Among all the measures, the runtime of the C_B measure is the highest.

The runtime trends in Fig. 7 indicate that the C_B measure rises sharply with network size *N*, unlike the other measures. Table VII emphasizes the three best-performing measures in bold.

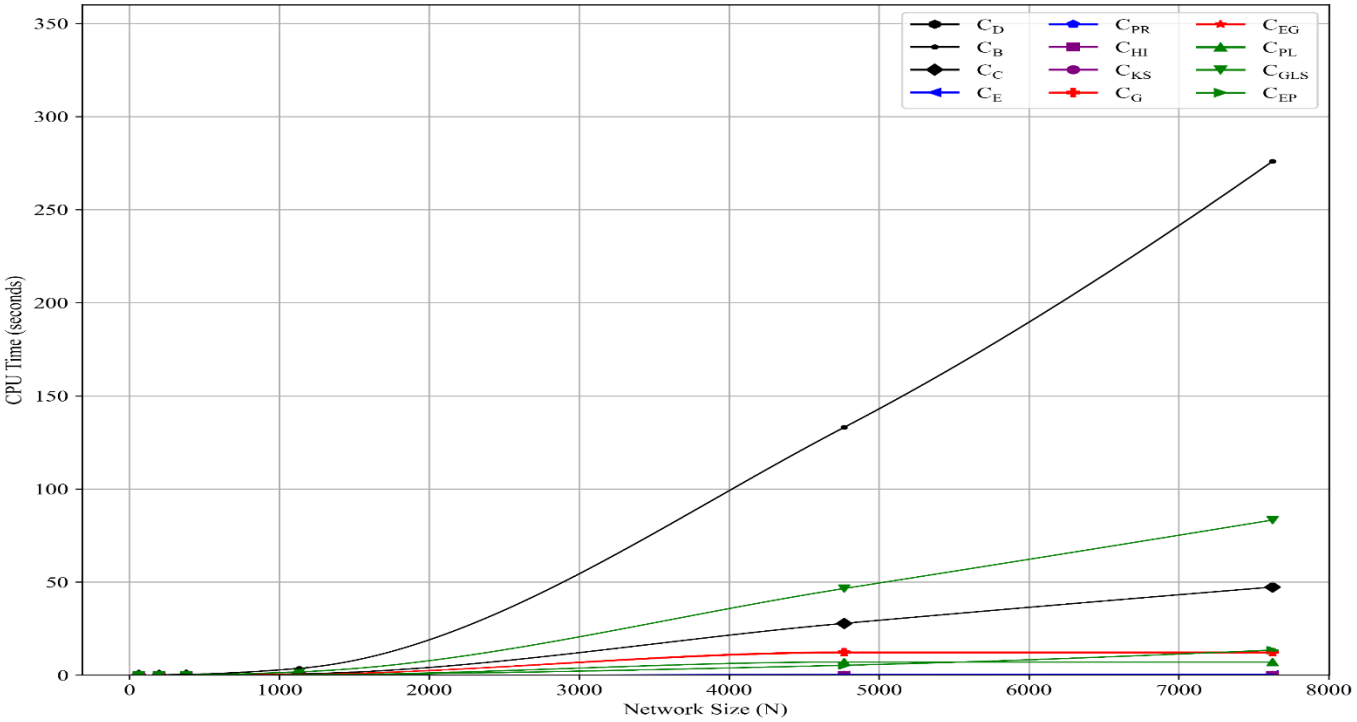


Fig. 7. The runtime (in seconds) of the 12 measures as a function of the Network size N.

TABLE VII. RUNTIME (IN SECONDS) OF 12 MEASURES

Network	C_D	C_B	C_C	C_E	C_{PR}	C_{HI}	C_{KS}	C_G	C_{EG}	C_{PL}	C_{GLS}	C_{EP}
LastFM	**0.0036**	350.6235	57.4717	0.6420	0.0914	**0.0726**	**0.0481**	4.7394	4.8089	1.5479	93.8552	21.0344
Power	**0.0035**	91.9589	11.3889	0.1699	0.0189	**0.0107**	**0.0108**	0.1051	0.1125	0.0870	24.3106	5.5635
Webspam	**0.0035**	186.5113	38.4280	0.2338	0.2421	**0.0457**	**0.0402**	14.7700	15.1397	8.9812	51.2695	6.6649
Dolphins	**0.00004**	0.0055	0.0014	0.0040	0.0018	**0.0002**	**0.0002**	0.0017	0.0017	0.0027	0.0093	0.0006

D. Discussion

Table VIII presents the findings of our research work, which indicates that the C_{EG} and C_G measures perform better than the other measures in every experiment except the run-time. Gravity-based measures have higher accuracy as compared to other measures because they are hybrid measures that provide both neighborhood and path information [34]. In most of the experiments, the C_{EG} measure attains the first position, and the C_G measure is at second because the C_{EG} measure includes a broader set of neighboring nodes as compared to the C_G measure. Table VIII depicts the three best-performing measures that outperform other measures.

TABLE VIII. COMPREHENSIVE COMPARATIVE EVALUATION TABLE

Network	Kendall's tau (τ)	F(t) vs Ranked Index	Monotonicity (M)	Runtime
LastFM	C_{EG}, C_G, C_{GLS}	C_{EG}, C_G, C_{KS}	(C_{EP}, C_{PR}, C_{EG} C_E) *, C_G, C_C	C_D, C_{KS}, C_{HI}
Power	C_{EG}, C_G, C_E	C_{EG}, C_G, C_{GLS}	(C_{EP}, C_E, C_{PR}) *, C_C, C_{EG}	C_D, C_{HI}, C_{KS}
Webspam	C_{EG}, C_E, C_{GLS}	C_{EG}, C_E, C_G	(C_{EP}, C_E) *, (C_{EG}, C_{PR}) *, C_G	C_D, C_{KS}, C_{HI}
Dolphins	C_{EG}, C_G, C_{EP}	C_{EG}, C_{EP}, C_G	(C_E, C_{PR}, C_G, C_{EG}, C_{EP}) *, C_{GLS}, C_{PL}	C_D, C_{HI}, C_{KS}

*Point out those measures that have gained the identical position.

V. CONCLUSION

This research has evaluated several centralities to pinpoint the significant nodes across diverse social networks. We measured the accuracy of several centralities by using the Kendall tau and SIR model among different social networks. In addition, we applied monotonicity to determine the differentiation ability of several centrality measures. It has been shown through empirical comparisons that the accuracy of both gravity-based measures, extended gravity centrality and gravity centrality, is superior to that of other centrality measures. Extended gravity measure emerges as the best performer, with the gravity measure ranking second. Furthermore, these gravity-based measures exhibit superior performance in terms of differentiation capability. However, both still require refinement to achieve higher efficiency in terms of runtime.

VI. Future Work

To increase the efficiency of the gravity-based measures, we will refine these measures in the future. This research work is limited to undirected unweighted networks; it can be extended to directed weighted networks to get deeper insights into the effectiveness of several measures.

Furthermore, the scalability and robustness of these measures can be assessed in dense networks. In order to improve the comprehensiveness of our analysis, we will also intend to incorporate new measures that have been built throughout this research.

References

[1] A. Zareie and A. Sheikhahmadi, "A hierarchical approach for influential node ranking in complex social networks," Expert Syst. Appl., vol. 93, pp. 200–211, 2018, doi: 10.1016/j.eswa.2017.10.018.

[2] A. Ullah, B. Wang, J. F. Sheng, and N. Khan, "Escape velocity centrality: escape influence-based key nodes identification in complex networks," Appl. Intell., vol. 52, no. 14, pp. 16586–16604, 2022, doi: 10.1007/s10489-022-03262-4.

[3] A. Ullah, B. Wang, J. Sheng, J. Long, and N. Khan, "Identification of Influential Nodes via Effective Distance-based Centrality Mechanism in Complex Networks," Complexity, vol. 2021, 2021, doi: 10.1155/2021/8403738.

[4] K. Das, S. Samanta, and M. Pal, "Study on centrality measures in social networks: a survey," Soc. Netw. Anal. Min., vol. 8, no. 1, 2018, doi: 10.1007/s13278-018-0493-2.

[5] S. P. Borgatti, "Centrality and network flow," Soc. Netw., vol. 27, no. 1, pp. 55–71, 2005, doi: 10.1016/j.socnet.2004.11.008.

[6] R. Grassi, S. Stefani, and A. Torriero, "Some new results on the eigenvector centrality," J. Math. Sociol., vol. 31, no. 3, pp. 237–248, 2007, doi: 10.1080/00222500701373251.

[7] L. C. Freeman, "Centrality in social networks conceptual clarification," Soc. Networks, vol. 1, no. 3, pp. 215–239, 1978.

[8] M. Kitsak et al., "Identification of influential spreaders in complex networks," Nat. Phys., vol. 6, no. 11, pp. 888–893, 2010.

[9] L. L. Ma, C. Ma, H. F. Zhang, and B. H. Wang, "Identifying influential spreaders in complex networks based on gravity formula," Phys. A Stat. Mech. its Appl., vol. 451, pp. 205–212, Jun. 2016, doi: 10.1016/j.physa.2015.12.162.

[10] K. Berahmand, A. Bouyer, and N. Samadi, "A new centrality measure based on the negative and positive effects of clustering coefficient for identifying influential spreaders in complex networks," Chaos, Solitons and Fractals, vol. 110, pp. 41–54, 2018, doi: 10.1016/j.chaos.2018.03.014.

[11] Z. Yu, J. Shao, Q. Yang, and Z. Sun, "ProfitLeader: Identifying leaders in networks with profit capacity," World Wide Web, vol. 22, pp. 533–553, 2019, doi: 10.1007/s11280-018-0537-6.

[12] A. Korn, A. Schubert, and A. Telcs, "Lobby index in networks," Physica A Stat. Mech. Its Appl., vol. 388, no. 11, pp. 2221–2226, 2009, doi: 10.1016/j.physa.2009.02.002.

[13] D. Chen, L. Lü, M. S. Shang, Y. C. Zhang, and T. Zhou, "Identifying influential nodes in complex networks," Phys. A Stat. Mech. Its Appl., vol. 391, no. 4, pp. 1777–1787, 2012, doi: 10.1016/j.physa.2011.09.017.

[14] Z. Li, T. Ren, X. Ma, S. Liu, Y. Zhang, and T. Zhou, "Identifying influential spreaders by gravity model," Sci. Rep., vol. 9, 2019, doi: 10.1038/s41598-019-44930-9.

[15] M. Bianchini, M. Gori, and F. Scarselli, "Inside PageRank," ACM Trans. Internet Technol., vol. 5, no. 1, pp. 92–128, 2005, doi: 10.1145/1052934.1052938.

[16] J. Sheng et al., "Identifying influential nodes in complex networks based on global and local structure," Phys. A Stat. Mech. its Appl., vol. 541, Mar. 2020, doi: 10.1016/j.physa.2019.123262.

[17] A. Ullah, S. U. Din, N. Khan, C. B. Mawuli, and J. Shao, "Towards investigating influencers in complex social networks using electric potential concept from a centrality perspective," Inf. Fusion, vol. 109, p. 102439, Sep. 2024, doi: 10.1016/j.inffus.2024.102439.

[18] A. Bavelas, "Communication patterns in task‐oriented groups," J. Acoust. Soc. Am., vol. 22, no. 6, pp. 725‐730, 1950.

[19] K. E. Joyce, P. J. Laurienti, J. H. Burdette, and S. Hayasaka, "A new measure of centrality for brain networks," PLoS One, vol. 5, no. 8, p. e12200, 2010.

[20] J. Bae and S. Kim, "Identifying and ranking influential spreaders in complex networks by neighborhood coreness," Phys. A Stat. Mech. its Appl., vol. 395, pp. 549–559, Feb. 2014, doi: 10.1016/j.physa.2013.10.047.

[21] A. Namtirtha, A. Dutta, and B. Dutta, "Identifying influential spreaders in complex networks based on kshell hybrid method," Phys. A Stat. Mech. its Appl., vol. 499, pp. 310–324, Jun. 2018, doi: 10.1016/j.physa.2018.02.016.

[22] J. Dai et al., "Identifying Influential Nodes in Complex Networks Based on Local Neighbor Contribution," IEEE Access, vol. 7, pp. 131719–131731, 2019, doi: 10.1109/ACCESS.2019.2939804.

[23] Q. Shang, Y. Deng, and K. H. Cheong, "Identifying influential nodes in complex networks: Effective distance gravity model," Inf. Sci. (Ny)., vol. 577, pp. 162–179, Oct. 2021, doi: 10.1016/j.ins.2021.01.053.

[24] J. Liu and J. Zheng, "Identifying important nodes in complex networks based on extended degree and E-shell hierarchy decomposition," Sci. Rep., vol. 13, no. 1, pp. 1–10, 2023, doi: 10.1038/s41598-023-30308-5.

[25] H. Singh and H. Singh, "An enhanced gravity model for determining crucial nodes in social networks based on degree k-shell eigenvector index," IEEE Access, vol. 12, pp. 23163–23180, 2024, doi: 10.1109/ACCESS.2024.3363635.

[26] A. Ullah, Y. Meng, and J. F. F. Mendes, "OVED-Rank: A ranking scheme to evaluate complex network spreaders' influence through the concept of effective distance and orbital velocity," Inf. Process. Manag., vol. 62, no. 5, p. 104201, Sep. 2025, doi: 10.1016/j.ipm.2025.104201.

[27] L. Qiu, J. Zhang, and X. Tian, "Ranking influential nodes in complex networks based on local and global structures," Appl. Intell., vol. 51, pp. 4394–4407, 2021, doi: 10.1007/s10489-020-02132-1.

[28] J. E. Hirsch, "An index to quantify an individual's scientific research output," Proc. Natl. Acad. Sci., vol. 102, no. 46, pp. 16569–16572, 2005.

[29] R. Xu and Y. Du, "A delayed SIR epidemic model with saturation incidence and a constant infectious period," J. Appl. Math. Comput., vol. 35, pp. 229–250, 2011, doi: 10.1007/s12190-009-0353-3.

[30] Y. Gang, Z. Tao, W. Jie, F. Zhong-Qian, and W. Bing-Hong, "Epidemic spread in weighted scale-free networks," Chinese Phys. Lett., vol. 22, no. 2, p. 510, 2005.

[31] Y. Moreno, R. Pastor-Satorras, and A. Vespignani, "Epidemic outbreaks in complex heterogeneous networks," Eur. Phys. J. B-Condensed Matter Complex Syst., vol. 26, no. 4, pp. 521–529, 2002.

[32] D. G. Bonett and T. A. Wright, "Sample size requirements for estimating pearson, kendall and spearman correlations," Psychometrika, vol. 65, pp. 23–28, 2000, doi: 10.1007/BF02294183.

[33] D. Christensen, "Fast algorithms for the calculation of Kendall's τ," Comput. Stat., vol. 20, pp. 51–62, 2005, doi: 10.1007/BF02736122.

[34] H. Singh and H. Singh, "Analysis of different measures of centrality to identify vital nodes in social networks," in Advanced Network Technologies and Intelligent Computing, vol. 2090, A. Verma, P. Verma, K. K. Pattanaik, S. K. Dhurandher, and I. Woungang, Eds., Commun. Comput. Inf. Sci., 2024, pp. 101–115, doi: 10.1007/978-3-031-64076-6_8.

Neural Networks for Pest Diagnosis in Agriculture: A Global Literature Review

Heling Kristtel Masgo Ventura[1]*, Italo Maldonado Ramírez[2], Roberto Carlos Santa Cruz Acosta[3], Wilfredo Ruiz Camacho[4], Juan Eduardo Suarez Rivadeneira[5], José Celso Paredes Carranza[6], Mayra Pamela Musayón Díaz[7], Cesar R. Balcazar Zumaeta[8], Carlos Luis Lobatón Arenas[9], Juan Alberto Rojas Castillo[10], Eli Morales-Rojas[11]*

Facultad de Ingeniería de Sistemas y Mecánica Eléctrica, Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas; Jr. Libertad N 1300, Bagua, Amazonas, Perú[1, 2, 3, 4, 5, 7, 9, 10, 11]

Facultad de Ciencias de la Salud, Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas; Calle Higos Urco N° 342, Chachapoyas, Amazonas, Perú[6]

Instituto de Investigación-Innovación y Desarrollo para el Sector Agrario y Agroindustrial (IIDAA), Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas, Chachapoyas 01001, Perú[8]

***Abstract*—Agricultural pests severely reduce global crop yields. To mitigate these losses, pest identification systems based on artificial intelligence have gained importance. This review analyzes worldwide advances in the use of neural networks for agricultural pest diagnosis, covering studies from 2007 to February 2024 retrieved from the Scopus database. Data were processed in Minitab 19 and spreadsheets, and keywords were mapped with VOSviewer. Results show that India and China lead scientific output, with research focused on corn, tomato, rice, and wheat. The most common architectures are ResNet, YOLO, and VGG-16/19, achieving performance metrics of up to 99 %. The review highlights the strong relationship between economic development and the adoption of neural networks. These findings provide researchers, agricultural engineers, and policymakers with a global perspective to guide future AI-based pest management strategies and support automation, especially in developing countries.**

Keywords*—Neural networks; pests; agriculture; developing countries

I. Introduction

Agriculture is fundamental to human health and population growth [1]. The world population is expected to reach 9.7 billion by 2050; therefore, food production must increase by 70% by 2050 [2]. But food production suffers from the environmental impact of chemicals [3]. In the world, the most important crop is soybean, and it is estimated that the crop has occupied 6% of the world's arable land since the 1970s [4].

Likewise, during the last decades, crops such as rice have become a staple food consumed by the majority of people around the world [5]. Asia is the center of rice production and produces more than 90% of the world's rice [6]. Another important commodity is maize, and it is widely grown [7], globally. 13% of malnourished children and 900 million poor households prefer maize as a staple food [8]. On the other hand, in the horticultural industry, tomato is recognized worldwide as one of the most cultivated vegetables and has a high nutritional value [9].

The increasing global demand for food production poses significant challenges to farmers in protecting their crops from harmful pests [10], with pests destroying up to 40% of global crop production each year [11]. Farmers suffer huge economic losses as income depends on the number of healthy crops they produce [12]. So, overcoming this problem becomes a major challenge, as agriculture is the most important economic branch in many countries [13].

In recent times, the application of artificial intelligence (AI) has become widespread in a number of areas, most notably its role in plant pest and disease identification [14]. Intelligent deep learning (DL) techniques have gained great popularity and have been widely adopted, especially in situations where human work cannot provide the speed and efficiency needed to analyze data on time and cover large areas in the field of monitoring [15].

AI and DL, especially image processing and convolutional neural networks (CNNs) are effective tools to apply in various tasks within the agricultural industry, such as leaf counting, leaf segmentation, and yield prediction [16], to enable farmers to effectively deal with plant leaf stress [17]. In recent agricultural research, techniques based on AI, DL, and CNN have shown great promise [10]. For example, in pest management for coconut crops [18], the deep learning model, VGG16, showed high precision and accuracy in diagnosing diseases, contaminated leaves, and insect infestation, demonstrating the potential for early disease detection. In studies to detect pests in maize [10], the MobileNet-SSD-v2 deep learning model was used with an overall relative error rate of 0.1579, which demonstrates high potential in real-time pest monitoring. The PlaNet model used in research [19] to diagnose diseased and healthy leaves achieved 97.95% accuracy. For pakcoy pest management [20] used known convolutional neural networks, such as MobileNetV2, GoogLeNet, and ResNet101, where the accuracy rate of tests reached 98 % as in the research of [9] that identified pests in tomato, using a convolutional neural network model based on GoogleNet, AlexNet, and ResNet-50c, and obtained an accuracy of 96.99%. While in the research of [21], they proposed a lightweight and effective agricultural pest detection model for small pests, called YOLO-Pest, which achieved 91.9% detection accuracy. On the other hand, research to identify agricultural pests [22] adopted the latest developments YOLOv3, YOLOv3-Tiny, YOLOv4, YOLOv4-Tiny, YOLOv6, and YOLOv8 for detection, where YOLOv8

*Corresponding authors

achieved a substantial mean average precision (mAP) of 91.9% in pest diagnosing.

Although several studies reported high accuracy and even strong mAP values, these metrics must be interpreted with caution. Accuracy alone does not capture whether infestations are systematically missed (low recall) or whether false alarms lead to unnecessary pesticide use (low precision). Therefore, integrating precision, recall, and mAP in model evaluation is essential to assess their real-world applicability in agriculture.

Based on the above, the objective of this research was to determine the advances of neural networks for pest diagnosis in agriculture through a global literature review.

The remainder of this paper is organized as follows. Section II describes the materials and methods, including the literature search strategy, data processing, and keyword analysis. Section III presents the results and discussion, highlighting global trends, key crops, and the most frequently applied neural network architectures for pest diagnosis. Section IV provides the main conclusions and outlines recommendations for future research and the practical adoption of artificial intelligence in agricultural pest management.

II. Materials and Methods

The bibliography consulted goes back to 2007, up to February 2024. Boolean operators were applied, using the following terms: "neural networks", "pests" and "agriculture". All the research was carried out through a search in the Scopus database, due to its capacity to compile open access texts, after a rigorous peer review [23] and 289 scientific articles were found and 129 studies were rescued and used for this study (Fig. 1).

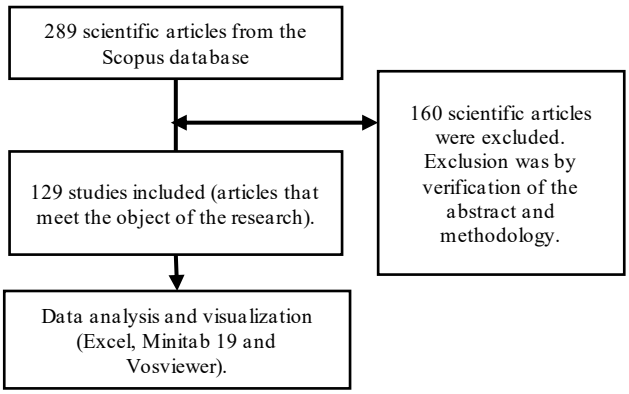


Fig. 1. Flowchart of scientific literature selection.

Publications from 2007 to February 2024 in all languages were considered. Titles, abstracts, methodology and main results were reviewed to select articles of interest. As well as the geographical scope was worldwide. Papers, such as book chapters, conference papers, and letters to the editor were excluded. Gray literature was also excluded because it did not pass peer review [24]. In addition, inconclusive studies and duplicates were not taken into account (Fig. 2).

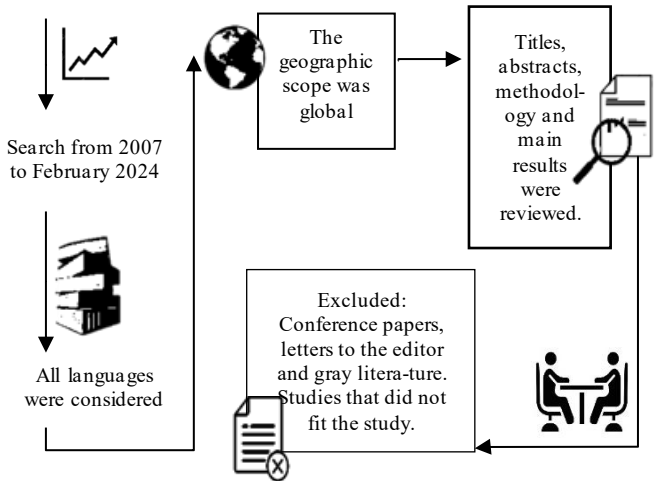


Fig. 2. Exclusion and inclusion process.

A. Data Analysis

The data were downloaded in CSV format and processed in Minitab 19 and spreadsheets to facilitate the determination of the distribution of studies by year and country. Keyword analysis was performed with VOSviewer version 1.6.19, a tool widely used in the scientific community to represent and visualize bibliometric networks. VOSviewer employs several colors to help understand and discover keyword relationships [25].

III. Results and Discussion

Fig. 3 shows the distribution of articles according to affiliation and country of origin, showing that the country's leading the studies related to the application of neural networks in agriculture are India (40 articles) and China (35 articles). These results are related to the advances in agriculture in India where approximately 62% of the population lives in rural areas and depends directly or indirectly on agriculture, being the main source of income. India's agricultural sector contributes almost 18% of India's GDP and ranks second in the world in production of agricultural products [26, 27]. However, the scarcity and lag in data availability in developing economies have necessitated artificial neural network (ANN) modeling techniques for price prediction in developing economies [28]. Likewise, in China, the agricultural industry generates jobs for more than 300 million farmers, and in recent years, the use of artificial neural networks (ANNs) has been applied in the agricultural sector [29–31]. In developing countries, agriculture plays a key role in the economy and provides rural inhabitants with higher incomes and job opportunities [32].

The evolution of publications per year is also evident; the growth occurred since 2016, highlighting 45 articles in the year 2023 (Fig. 4A), evidencing a breakthrough in the application of deep learning techniques to the identification of plant diseases and pests, given that they are one of the greatest threats to food security [33, 34].

The largest amount of scientific production is focused on research on corn, tomato, rice, wheat, apple, citrus, grape, and cotton crops are the crops that predominate in neural network studies for pest diagnosis in agriculture (Fig. 4B).

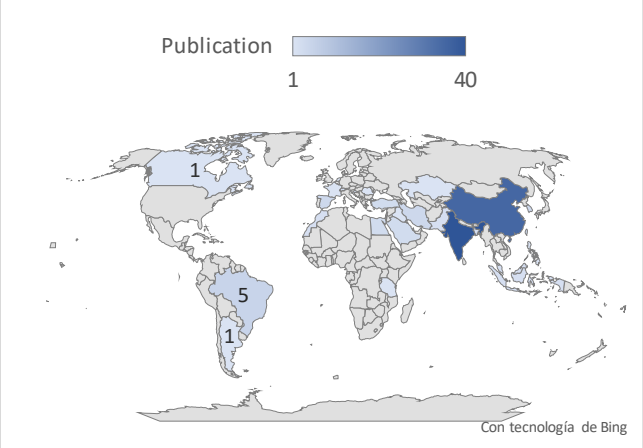


Fig. 3. Distribution of scientific production by country.

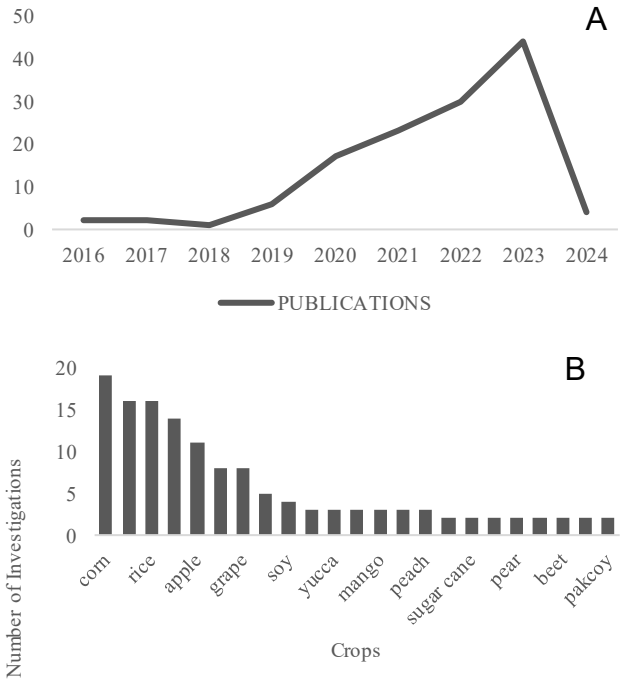


Fig. 4. Evolution of publications with respect to agricultural crops.

In Fig. 5, it is evident that the three most used neural network architectures in the explored studies are ResNet and versions, YOLO and versions, and VGG-16, 19. The neural network architectures with GoogleNet were the ones that evidenced fewer studies, although these are based on the construction of a deeper model to achieve greater accuracy and, at the same time, keep it computationally efficient [35].

Beyond their frequency of use, each architecture offers specific advantages that explain their adoption in agricultural applications. ResNet is valued for its ability to train deep networks effectively while mitigating vanishing gradient issues, making it suitable for complex image classification tasks [9, 49, 85, 165]. YOLO stands out for its speed and real-time object detection, which is essential for pest monitoring directly in the field [21, 22, 55, 70]. VGG-16/19, although computationally heavier, remains relevant for its high accuracy and robustness in image recognition [9, 49, 124]. GoogleNet, with its inception modules, provides efficiency with fewer parameters, which is particularly advantageous in environments with limited computational resources [9, 35]. These differences help explain why researchers select certain models depending on the crop, available resources, and the practical requirements of the study.

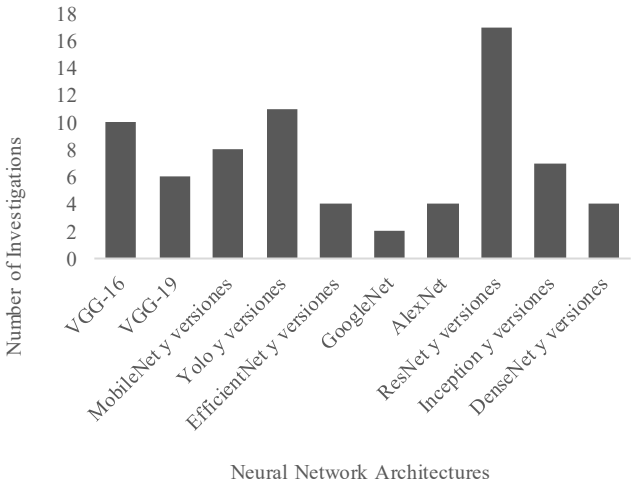


Fig. 5. Known neural network architectures used in research.

The keyword co-occurrence is related to the emerging theme regarding global agriculture, with the keywords "Convolution neural networks, image processing, and Deep learning (Fig. 6). The network of co-occurrence of words indicates the relationship of the emerging studies in the world, due to the interest in improving agricultural productivity and promoting economic growth through nondestructive alternative techniques in pest recognition [36–38].

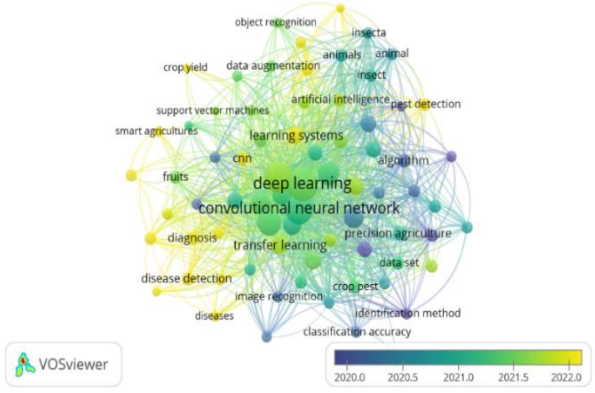


Fig. 6. Map of the keyword concurrence network.

Beyond the bibliometric visualization, these keyword clusters reflect concrete agricultural challenges. The prominence of terms such as deep learning and convolutional neural network highlights the reliance on high computational resources and large annotated datasets [30, 38]. However, such requirements are difficult to meet in developing countries, where data scarcity, limited infrastructure, and high implementation costs restrict practical adoption [13, 14]. In these contexts, farmers often cannot afford advanced computing systems or generate

sufficient labeled images, which limits the transfer of research advances to the field [29,171]. Therefore, the co-occurrence network not only maps research priorities but also underscores the urgent need for cost-effective, data-efficient, and locally adaptable AI models to ensure wider adoption in agriculture.

Table I shows the related studies according to the quartile of the journal models/Neural Network Architectures, crops evaluated, and the performance metrics of the models, where it is evident that the studies are focused on neural networks to detect pests and diseases in maize crops, based on improved GoogLeNet and Cifar10 models for the recognition of leaf diseases [39]. The highest recognition accuracy found was 95.3%; however, it is limited to the number of diseases in maize [40]. As well as its evaluation yield of each year of corn, will serve as a basis for making accurate decisions about harvesting and marketing of corn grain in real time, minimizing possible losses of profitability.

Tomato is one of the most popular and appreciated vegetables among Asians, and worldwide, it is the second most consumed vegetable [41]. Moreover, tomato is not only served as a vegetable, but also serves as a sauce, jam, etc. and is used in the processing of different types. But it is affected by different pests; in that sense, to identify tomato pests, have selected image datasets from repositories there are convolutional neural network (CNN) models to deal with this problem [42]. In this regard, several studies using images of common tomato pests have classified pest categories [43]; therefore, continuous monitoring is necessary for early disease detection [44]. Another crop widely studied by CNN models is rice and wheat; this may be related to the fact that rice is one of the most important agricultural products in the world; this crop is the staple food for more than half of the world's population [45, 46]. Approximately 160 million hectares are planted annually and produce 750 million tons of rice [47]. Given the scarcity of water for agriculture, the increase in food demand, and future drought scenarios, it is essential to design new technologies that contribute to lower water consumption [48].

On the other hand, we have the application of Model Fuzzy Modified Faster Fuzzy Region based CNN (MGAN MFRCNN with Fuzzy) on leaves of diseased and healthy banana plants [49] attacked by banana leaf spot disease [50]. It is important to predict leaf disease symptoms at an early stage and to develop an automatic detection technique. Considering that India is at 19% production, followed by Brazil with 15% and Ecuador with 12%. China produces 10% and the review shows only one research, which indicates that studies are just beginning to focus on this important sector.

For the assessment of potato leaf diseases, several machine learning techniques have been developed, among them is the multilevel deep learning model, where the potato leaves are extracted from the potato plant image using the YOLOv5 image segmentation technique. Where it is extracted, the potato leaves from the potato plant image are used using the YOLOv5 image segmentation technique. In this study, the widely used deep learning hierarchical CNN (HDLCNN) model is evidenced for data sets of diseased and healthy potato plant leaves. This is because of the great trend, given the crops are affected by various diseases caused by pests and pathogens such as viruses, bacteria and fungi [51], hence, the material used for this type of work is images taken from healthy and infected leaves of the plant and the accuracy has reached up to 98.9% with no signs of overfitting [52].

It was found that the CNN model was also used to identify major diseases in grape crops. As well as an automatic method to monitor pests based on a CNN with a dataset of 177 images with apple moth, however, researchers mention that CNN models have several difficulties in identifying crop diseases due to morphological and physiological changes in crop tissues and cells, because some studies already report a lightweight CNN model called GrapeNet for identification of different symptom stages of specific grape diseases [53].

TABLE I GENERAL CHARACTERISTICS OF THE SCIENTIFIC PRODUCTION NEURAL NETWORKS FOR PEST DIAGNOSIS IN AGRICULTURE

Quote	Quartile	Neural Network Model	Dataset	Crop(s)	Pest(s)/Disease(s)	Performance Metrics
[18]	Q1	VGG-16	Kaggle	Coco	Whitefly	Accuracy: 95.71%
[54]	Q1	VRFNet (Visual Regenerative Fusion Network)	D0, IP102	Multiple crops	Insects (varied age, size, shape, color)	Accuracy: D0 = 99.12%; IP102 = 68.34%
[10]	Q2	MobileNet-SSD-v2-Lite	2,605 images	Corn crops	Ladybugs, beetles (Coccinella sp., Anoxia villosa)	mAP: 0.8923; Relative error: 15.79%
[55]	Q1	Improved Pest-YOLO	Pest24 (25,378 images)	Leaves of various crops	24 pest classes	mAP: 73.4%; Recall: 83.9%
[56]	Q2	FDPRC-Net (Feature Pyramid Dilation Residual CNN)	Mixed (Wang, Xie, Tomato pests)	Tomato	Mixed pest classes	Accuracy: Tomato = 98.12%, Wang = 97.43%, Xie = 93.98%, Overall = 93.46%
[57]	Q3	CNN (ABC-CNN + Adam optimizer)	Kaggle	Tomato	Key tomato pests	Accuracy: 99.33%; MAE: 0.007; MSE: 0.007
[19]	Q1	CNN (PlaNet)	PlantVillage, Kaggle	corn, apple, grape, etc.	Leaf diseases (spot, rust, scab)	Accuracy: 97.95%; AUC: 0.9752; F1-score: 0.9686
[58]	Q1	HCNet (Hierarchical Complementary Network)	IP102	—	Diverse insect species	Accuracy: 75.36%

[59]	Q4	CNN (new proposed model)	19,046 images	Rice and wheat	Green caterpillar, A. Tridens, rice bug, etc.	Accuracy: 99%
[60]	Q1	CSLSNet (proposed CNN model)	PlantVillage	Tomato	Early blight, mosaic virus, yellow virus	Accuracy: 90.08%
[20]	Q3	MobileNetV2	1,226 labeled images	Pakcoy cultivation	Leafminers, cabbage butterflies, powdery mildew	Accuracy: 98%
[61]	Q1	Cotton LeafNet (CNN-based)	22 types of cotton leaf disease images	Cotton crops	Leaf diseases, bacteria, fungi, viruses, nutrient deficiency	Accuracy: 99.39%
[62]	Q1	ITF-WPI intermodal feature fusion model	10,598 image and text samples	Goji berry	Wolfberry pests (WPIT9K)	Accuracy: 97.98%, F1-Score: 93.19%
[63]	Q1	RetinaNet with R50-FPN and R101-FPN	400 melon leaf images	Melon leaves	Leafminers	mAP: 92.36%, Recovery Rate: 92.70%
[64]	Q2	MGAN-MFRCNN (CNN + Fuzzy logic)	Healthy and diseased banana leaf images	Banana crops	Xanthomonas, Sigatoka	Accuracy: 98%, F1-Score: 96%
[65]	Q1	LSTM-CNN	4,447 pest and disease instances	Apple	Apple pests and diseases	Accuracy: 99.2%
[66]	Q2	Mask R-CNN	Fall armyworm (FAW) insect dataset	Corn	Fall armyworm detection	mAP: 94.21%
[67]	Q2	HGS-DCNN (Optimized CNN with preprocessing)	Augmented pest dataset with variable ages, colors, etc.	Various	Insect detection across variability	Accuracy: 99.1%, F1-Score: 97.80%
[68]	Q3	MIL-CNN (Multi-instance learning CNN)	12,000 images	Cotton crops	Whitefly detection	Accuracy: 98.13%
[51]	Q1	HDLCNN (Hierarchical Deep Learning CNN)	Dataset of diseased and healthy potato leaves	Potato	Major potato diseases	Accuracy: +4%; Precision: +6%; Recall: +3%; F1-score: +3.5%; Specificity: +4.5%; Sensitivity: +1%; PSNR: +2% (vs. VGG-INCEP, Deep CNN, RF, SNN)
[69]	Q3	Improved YOLOv5 (lightweight algorithm)	Dataset with 15 species of agricultural pests	Various crops	Insects (various ages, colors, shapes, sizes)	Accuracy improved by 4.3% over YOLOv5n; mAP@0.5: 95.3%
[70]	Q1	YOLOv5s (basic architecture)	7737 images from IP102 dataset	Leaf crops	Insects (various ages, colors, shapes, sizes)	Accuracy: 98.1%; Recall: 97.5%; mAP@0.5:0.95 = 88.1%
[71]	Q1	EfficientNetB7 and VGG16	600 leaf images from different species	Red beans, black beans, mango, cranberry, chickpea, lima bean, soybean	Healthy and diseased leaves	Accuracy: 96%–98%
[72]	Q1	Perceptron + Fuzzy Logic; ANN; CBR	~600 images	Sugarcane	Eye spot, leaf scald, yellow leaf, pokkah boeng	Eye spot: SEN 85.12%, SPEC 84.96%, Acc 83.72%; Scald: SEN 85.16%, SPEC 84.85%, Acc 83.42%; Yellow leaf: SEN 85.95%, SPEC 84.26%, Acc 83.41%; Pokkah Boeng: SEN 85.76%, SPEC 84.28%, Acc 83.72%
[73]	Q1	Mask R-CNN R50 FPN3; Fast R-CNN; InceptionV3	1000 image samples	Olive	Fungi and *Aceria oleae*	Training accuracy: 90%; Validation accuracy: 85%
[9]	Q3	GoogleNet, AlexNet, ResNet-50	PlantVillage dataset	Tomato	Bacterial spot, early blight, leaf mold, Septoria leaf spot	Accuracy: 96.99%

[74]	Q2	Smart Farm Software with Sigmoid Sorting	900 training images, 890 test images	Rice	*Chilo suppressalis*	Accuracy: 88%–92%; Recall: 91%
[75]	Q1	Three-Scale Care CNN (TSCNNA)	21,000 pest images	Corn, cabbage	Corn borer, cabbage moths, larvae, cabbage blight	Accuracy: 93.16%
[76]	Q3	15-layer Augmented CNN	Kaggle rice dataset	Rice	Leaf charcoal, bacterial leaf blight, brown spot	Accuracy: 95%
[77]	Q4	Decision Tree (DT), CNN, ResNet, Attention-based CNN	Dataset of 18,000 images	Common plants	Whitefly	DT: 81%, CNN: 96%, ResNet: 97.5%, Attention-CNN: 98% (Accuracy)
[78]	Q2	Deep Neural Network (DNN)	Dataset of 430 images	Apple trees	Apple moth	Accuracy > 99%
[22]	Q2	YOLOv3, YOLOv3-Tiny, YOLOv4, YOLOv4-Tiny, YOLOv6, YOLOv8	Dataset of 9,875 images	Various crops	Thistle caterpillars, red beetles, citrus psylla	YOLOv8: mAP = 84.7%, Loss = 0.7939
[79]	Q1	YOLOv3 Optimized, ResNet50, VGG16	IP102 dataset	Various crops	Multiple pests including rice leaf caterpillar, rice leaf roller, rice stalk fly	Accuracy = 96%, F1 Score = 84%
[80]	Q2	GPA-Net (Pyramidal CNN with Graphical Attention)	IP102 dataset	Cassava leaves	Multiple agricultural pests	Accuracy = 99%
[81]	Q1	VGG19 Classifier	Dataset of 862 images	Various crops	Codling moth larvae (Spodoptera frugiperda)	Accuracy = 99%
[21]	Q1	YOLO-Pest	Teddy Cup dataset and IP102	Various crops	Creatonotus, Nilaparvata, Staurophora celosia	mAP@0.5 = 91.9%
[82]	Q1	Custom CNN model	Dataset of 5,000 images	Potato crops	Sana, black dandruff, scabies, blackleg, pink rot	Accuracy = 99%–100% across disease classes
[83]	Q3	MLP Neural Network	Dataset of 300 images	Apple tree leaves	Black spot, Alternaria, Minoz blight	CC index = 0.976, RMSE = 0.098
[84]	Q1	Multi-image fusion recognition method	IP102, DO, and ETP datasets	Rice, corn, wheat, beet, alfalfa, citrus, tomato, mango	Leaf roller, caterpillar, wireworm, bactrocera tsuneonis	Accuracy = 88.7%
[85]	—	R-CNN, ResNet	Full dataset (images & videos of diseased and healthy leaves)	Tea, apple trees	Leaf diseases	Accuracy = 99.2%
[86]	Q3	RDODL-APDC, NestNet, MobileNet-v3	Dataset of 7,222 grape and 7,771 apple disease images	Apple, grape plants	Scab, Black Rot, Rust, Cedar Apple, Leaf Blight	Apple disease accuracy = 95.8%, Grape = 97.19%
[87]	Q1	EfficientNetV2	PlantVillage, IP102	Various crops	Leaf spots, rust, late blight, cucumber mosaic virus, tomato mosaic virus, etc.	Accuracy = 99.71%
[88]	Q4	ResNet50	1,221 images	Grenada crops	Bacterial blight, anthracnose, fruit spot, fusarium wilt, fruit borer	Accuracy = 98.55%
[89]	Q4	VGG16, ResNet50, AlexNet, EfficientNetB2, EfficientNetB3	41,763 images	Tomato leaves	Tomato pests and diseases	Accuracy = 99.85%
[90]	Q2	Enhanced CNN (VGG16-based)	1,003 wheat images	Winter wheat	Aphid, powdery mildew, leaf rust, linear rust	Accuracy = 96.02%

[91]	Q1	YOLOv7	IP102	Corn	Corn borer, budworm	mAP@0.5:0.95 = 96.69%, Accuracy = 99.95%
[92]	Q2	Hybrid CNN + GAN	Xie2	Various crops	Green-horned caterpillar, stink bug, Helicoverpa, etc.	Performance gain: AlexNet = +3.75%, ResNet50 = +2.74%, ResNet101 = +1.54%, GoogleNet = +1.76%, VGG16 = +1.76%, VGG19 = +2.74%, Simple CNN = +2.14%
[93]	Q2	ANN (Model 1: PMD, Model 2: VMD)	400 images	Apple trees	Blister moth	Model 1: Accuracy = 98%, Model 2: Accuracy = 94%
[1]	Q3	VGG16, VGG19, InceptionV3, MobileNetV2-mod	Corn image database (augmented)	Corn	Budworm	VGG16 = 96.17%, VGG19 = 97.15%, InceptionV3 = 99.23%, MobileNetV2-mod = 99.13%
[94]	Q1	YOLOv5s (modified)	1,565 images	Various crops	Ants, grasshoppers, palm weevils, shield bugs, wasps	Precision = 0.018, Recall = 0.015, mAP = 0.011
[95]	Q2	Custom CNN (convolution + clustering, batch normalization layers in series and parallel)	598 citrus images (Citrus Leaves Prepared)	Citrus leaves	Black spot, canker, greening	Accuracy = 96%, F1-score = 95%, Precision = 96%, Recall = 95%
[96]	Q1	CNN with fine-tuned ResNet50	1896 images of oranges	Orange	Black spot	Accuracy = 99.5%, F1-score = 100%
[97]	Q1	ADM (Anomaly Detection Model), DIM (Disease Identification Model), LPDM (Leaf Powder Distinction)	~9000 tomato leaf images	Tomato leaves	Leaf mold and powdery mildew	ADM Acc = 97.4%, DIM Acc = 93.63%, LPDM Acc = 98.7%
[98]	Q1	VGG19	2892 rice leaf images (Kaggle dataset)	Rice	Hispa, brown spot, leaf blight, NPK deficiency	Accuracy = 91.8%
[99]	Q1	Fine-tuned Inception-v3	IP102 dataset	Various crops	Parasitic insect pests (e.g., rice leaf caterpillar, rice roller, Apolygus lucorum, etc.)	Accuracy = 67.88%
[100]	Q1	Hybrid CNN (transfer learning and fine-tuning)	PlantVillage dataset	Tomato crops	Bacterial spot, early blight, late blight, leaf mold, Septoria leaf spot, spider mites, target spot, TYLCV	Accuracy = 98.1%
[101]	Q2	OplusVNet (CNN13 + VGG16)	2071 citrus images	Citrus	Canker, leaf miner, rust scab, rusty wall, citrus scab, etc.	Accuracy = 99%
[102]	Q1	Optimized MobileNetV2	IP102 dataset	Various crops	Insect pests (e.g., rice leaf caterpillar, rice roller, Apolygus lucorum, etc.)	Accuracy = 71.32%
[103]	Q2	Faster R-CNN, Mask R-CNN, YOLOv5	Baidu AI insect dataset, IP102	Rice, wheat, corn	Military armyworm, Asian rice borer, brown planthopper, rice borer, English grain aphid, rice gall midge	YOLOv5 >99%, Faster/Mask-RCNN >98%
[104]	Q2	New CNN model built from scratch	Plant Village dataset	Grape crops	Main grape diseases	Accuracy = 99.34%; F1 score = 0.9934
[105]	Q2	Hybrid deep learning model	Dataset with most common crop pests	Various plants	Insects of different ages, colors, sizes, and shapes	SSIM = 0.99; MAE < 0.2; AP = 89.67%
[106]	Q2	CNN with image mosaic	Dataset of 58,349 images of beetle-bitten leaves	*Brassica chinensis*	Flea beetle (*Phyllotreta undulata*)	99.7% detection of bitten leaves

[107]	Q3	Deep learning architecture with minimal parameters	Kaggle dataset	Yucca plants	Cassava bacterial blight (CBB), brown streak (CBSD), cassava green mite (CGM), cassava mosaic (CMD)	Accuracy = 90%
[108]	Q2	Classic ResNet50 network	Dataset of 900 images	Corn husks	*Spodoptera frugiperda*	Validation accuracy: ResNeSt50 = 98.77%, ResNet50 = 97.59%, EfficientNet = 97.89%, RegNet = 98.07%
[109]	Q2	CNN with majority voting ensemble and early merger ensemble	Turkey PlantDataset (4,447 images)	Apples, peaches, pears, cherries, etc.	*Aphis spp.*, *Eriosoma lanigerum*, *Monilia laxa*, drying symptom, *Parthenolecanium corni*, *Erwinia amylovora*	Accuracy: Majority voting = 97.56%, Early merger = 96.83%
[17]	Q1	Two-stage CNN: semantic segmentation + symptomatic lesion classification	BRACOL dataset	Coffee leaves	Leaf miner, brown leaf spot, *Cercospora* leaf spot, rust, general leaf spot	Accuracy > 97%
[110]	Q1	Faster R-CNN with MobileNetV3 backbone	Dataset of 36,000 images	-	*Popillia japonica*, *Cetonia aurata*, *Phyllopertha horticola*	Accuracy = 92.66%
[111]	Q3	Tuned CNN	Dataset of 4,868 images	Cucumber leaves	Spider, leaf miner, downy mildew, powdery mildew	Accuracy = 98.19%
[112]	Q4	Ensemble CNN (VGG-19, ResNet-50, InceptionV3)	Dataset of 18,345 images	Tomato and cotton	Bacterial spot, early blight, late blight, leaf mold, septoria leaf spot, bacterial blight, curly top virus, *Fusarium* wilt	Accuracy = 97.9%
[113]	Q3	Ensemble Learning (VGG16 + VGG19 + Xception, transfer learning)	PlantVillage (grape leaves)	Grape	Black Rot, Black Measles (Esca), Leaf Blight	Accuracy: 99.82%; Precision/Recall/F1 = 1.00
[114]	Q1	CNN + Transfer Learning (ResNet50, DenseNet121, InceptionV3)	Custom image dataset (field-collected tomato leaves)	Tomato	Early blight, Late blight, Septoria leaf spot	Accuracy: 98.7%; Precision: 98.5%; Recall: 98.6%; F1-score: 98.5%
[115]	Q2	Modified Capsule Network (MCapsNet)	~2000 images	-	Mucolycid worms, corn borers, moths, caterpillars, ladybugs, aphids, cotton bollworms, cicadas	Accuracy = 87.52%; Recovery = 78.30%
[5]	Q2	RDD_CNN Model	4398 images	Rice leaves	Brown spot disease, bacterial blight, stemborer	Accuracy = 98.47%
[116]	Q2	Enhanced EfficientNet	IP102 dataset	-	Various pests	Accuracy = 69.45%; F1 score = 63.06
[103]	Q1	Multi-branch CNN (Mb-CNN)	1100 images (aphids)	Wheat, corn, rapeseed	Aphids	MAE = 10.22; MSE = 12.24
[117]	Q4	ResNet50, ResNet18, Inception-V3	1493 images	Pomegranate	Bacterial blight, anthracnose, fruit blotch, wilt, fruit borer	ResNet50 = 97.92%; ResNet18 = 87.5%; Inception-V3 = 78.75%
[118]	Q4	Super-Resolution CNN (SRCNN)	54,343 images	Apples, tomatoes, grapes, corn, potatoes	Apple scab, black rot, bacterial spot, citrus greening, etc.	Accuracy = 99.175%
[119]	Q2	Enhanced MobileNet-V2	3503 images (Kaggle)	Apple, cassava, corn, cotton	Rust, apple scab, cassava brown streak, corn rust, etc.	Accuracy = 92.20%

[12]	Q4	CNN segmentation model	2424 images	Tomato leaves	Tuta absoluta	Min. confidence = 70% in 5 seconds
[120]	Q1	Big data + deep learning	52,322 images	Pear leaves	Leucoptera malifoliella	Accuracy = 91.3–99.5%; F1 = 0.69–0.93
[121]	Q1	Mask R-CNN (ResNet backbone)	2500 images	Strawberry	Angular leaf spot, anthracnose, gray mold, powdery mildew, etc.	Accuracy = 82.43%
[122]	Q1	Lightweight CNN (SimpleNet)	568 images	Wheat	Glume spot scab	Accuracy = 94.1%
[123]	Q1	Faster R-CNN	IP102 dataset	-	Aphids, cicadellidae, linseed budworm, flea beetles, mites	Accuracy = 99.0%
[124]	Q2	Deformable VGG-16 (DVGG-16)	>2000 images	-	Rice borers, moths, caterpillars, bollworms, bugs, locusts	Accuracy = 91.14%
[125]	Q4	DL + RMSProp, Adam, SGDm	8000 images	Citrus crops	Snail infestation	Accuracy = 98.73%
[126]	Q1	InceptionResNetV2	Kaggle dataset	Rice leaves	Leaf blight, brown spot, bacterial blight	Accuracy = 95.67%
[127]	Q1	DNN-SAR (Optimization method)	2326 images	Rice	*Cnaphalocrocis medinalis*, *Scirpophaga incertulas*	Accuracy = 98.29%
[128]	Q1	GoogleNet and ResNet50	60,659 images	Cotton leaves	Cotton leaf diseases	GoogleNet = 86.6%; ResNet50 = 89.2%
[129]	Q2	Cascade approach CNN	Greenhouse image dataset	-	Flies, midges, thrips, whiteflies	F1 = 0.92/0.90; Count accuracy = 0.91/0.90
[130]	Q1	Deep learning + computer vision	841 images	Grape leaves	Mildew, spider mite	Accuracy = 94%; F1 = 0.94
[131]	Q1	Hand-designed CNN + MobileNet + InceptionResNetV2	1564 images	Coconut palms	Hemorrhagic stem disease, leaf blight, red palm weevil	CNN = 96.94%; MobileNet = 82.10%; InceptionResNetV2 = 81.48%
[132]	Q3	AlexNet	4344 images	Rice	Borer, brown leafhopper, leaf folder, green leafhopper	Accuracy = 96.9%
[133]	Q3	YOLOv5	5000 images	-	Ambrosia, amaranth, bromo	Accuracy = 82–92%
[134]	Q3	Custom DL model	3000 images in wheat fields	Weeds	*Capsella*, *Chenopodium*, *Sinapis arvensis*, *Tripleurospermum*	Accuracy = 98%
[135]	Q4	DenseNet201	859 images	Tomato crops	Major tomato pests	Accuracy = 94.87%
[136]	Q2	VGG-19-based model	3199 images	Peach	Multiple bacterial and fungal diseases	Accuracy = 94%
[137]	Q2	ResNet-50 tuning	3549 images	Rice	*Leptocorisa acuta*, *Locusta migratoria*, etc.	Accuracy = 95.01%
[138]	Q1	Tiny-YOLOv3	~5000 images	Longan	*Tessaratoma papillosa*	mAP = 89.72–95.33%
[139]	Q3	CNN architecture	Plant Village + Digipathos	Various crops	Multiple fungal and bacterial diseases	Accuracy = 99.85%
[140]	Q1	Seq-RNN	Plant Village	Bell pepper	Bacterial spot, other bell pepper diseases	Performance = 98.17
[141]	Q1	New CNN model	CPAF dataset, 73,635 images	-	Insects and larvae	Accuracy = 92.26%
[142]	Q1	Faster RCNN ResNet-50	IP102 + Bugwood	-	Diverse insects (e.g., ants, grasshoppers)	Accuracy = 94%
[143]	Q1	BridgeNet-19	12,561 images	Citrus	Citrus pests (e.g., psyllid, aphids, cicada)	Accuracy = 95.47%

[144]	Q1	ResNet-18 CNN	4753 images	-	*Drosophila suzukii* (fruit fly)	AUC = 0.506–0.603
[145]	Q3	DenseNet121-based CNN	Dataset of 66 pest species	-	66 pest species incl. *Spodoptera*, *Chilo*, *Leptocorisa*	ID rate = 93.9%; False alarm = 8.2%
[146]	Q1	Proposed CNN model	1600 images	Cotton	*Aphis gossypii*, *Anthonomus grandis*, *Helicoverpa* spp.	Accuracy = 98%
[147]	-	Proposed CNN	4300 images	Mango	*Capsid bug*, *Cecid fly*, *Fruit fly*, *Leaf hoppers*	Accuracy = 88.75%
[148]	Q1	ResNet50-based	1747 images	Coffee	Leafminer, rust, cercospora	Symptom class. >97%; Severity = 86.51%; Biotic rating = 95.24%
[149]	Q4	AlexNet	IP102 dataset	Multiple crops	*Xylotrechus*, *Ampelophaga*, etc.	Accuracy = 89.33%
[150]	Q3	DenseNet169	859 images	Tomato	*Tuta absoluta*, *Bactrocera*, *Bemisia tabaci*	Accuracy = 88.83%
[151]	Q3	R-CNN and Mask R-CNN	1239 images	Bell pepper	Whitefly, thrips, mites	R-CNN = 89%; Mask R-CNN = 81%
[152]	Q1	WRN (Wide ResNet)	>36,000 samples	Tomato, potato, grape, apple, corn	Typical crop diseases	Accuracy = 85–99%
[153]	Q1	New CNN architecture	>100,000 images	Wheat, rice, corn, barley, rapeseed	Various plant diseases	BAC = 0.98
[154]	Q1	CFN	2200 images	Wheat, corn, canola	Aphids	Accuracy = 76.8%
[155]	Q1	CNN (NBAIR + Xie1/Xie2)	Diverse datasets	Rice, wheat, corn, soy, sugarcane	10+ pests	Accuracy = 96.75%
[156]	Q1	Transfer Learning CNN	2 weed types	Weeds	*Sisymbrium sophia*, *Veronica persica*	Accuracy = 98.92%
[157]	Q1	Weakly DenseNet-16	12,561 images	Citrus	Medfly, psyllid, stink bug, canker	Accuracy = 93.33%
[158]	Q1	Residual neural system	8178 images	Wheat	*Septoria*, *Tan spot*, *Rust*	Balanced accuracy = 0.87
[159]	Q3	CNN (Real-time monitoring)	Grain pest dataset	Wheat	*Sitophilus oryzae*, etc.	Accuracy > 90%
[160]	Q1	CNN + augmentation	4400 images	Wheat, rice	*Sawfly*, *Aphid*, *Mite*, *Leafhopper*	mAP = 81.4%
[161]	Q2	CNN (Morphological analysis)	280 images	Rice	*Chilo spp*, *Gryllotalpa*, etc.	Accuracy = 81.82–94.44%
[162]	Q1	AlexNet, VGGNet, ResNet	10,000 images	Soybean	Defoliation estimation	MSE = 4.57
[163]	Q1	VGG16	687 images	Rice	*BPH (Brown grasshopper)*	Accuracy = 95%
[164]	Q1	Faster R-CNN, R-FCN, SSD	5000 images	Tomato	Leaf miner, mold, whitefly, etc.	FRCNN: 0.413–0.906; R-FCN: 0.7545–0.9492; SSD: 0.762–0.8841
[165]	Q1	Residual CNN	555 images	Corn, wheat, soy, canola	5 insect pests	Accuracy = 98.67%
[166]	Q1	CNN whitefly ID algorithm	3185 images	-	*Bemisia tabaci*, *Frankliniella occidentalis*	Precision: 0.92–0.96; F1: 0.94–0.95
[167]	Q1	CNN moth detector	177 images	-	Apple moth	AUC increased from 0.931 to 0.934

CNN: Convolutional Neural Network; VGG16: CNN with 16 depth layers; YOLO: You Only Look Once, real-time object detection; ResNet: Residual Neural Network; Inception: CNN with multi-scale filters; EfficientNet: Scalable CNN; MobileNet: Lightweight CNN; DenseNet: Densely Connected CNN; Mask R-CNN: Region-based CNN for detection and segmentation; LSTM: Long Short-Term Memory; GAN: Generative Adversarial Network.

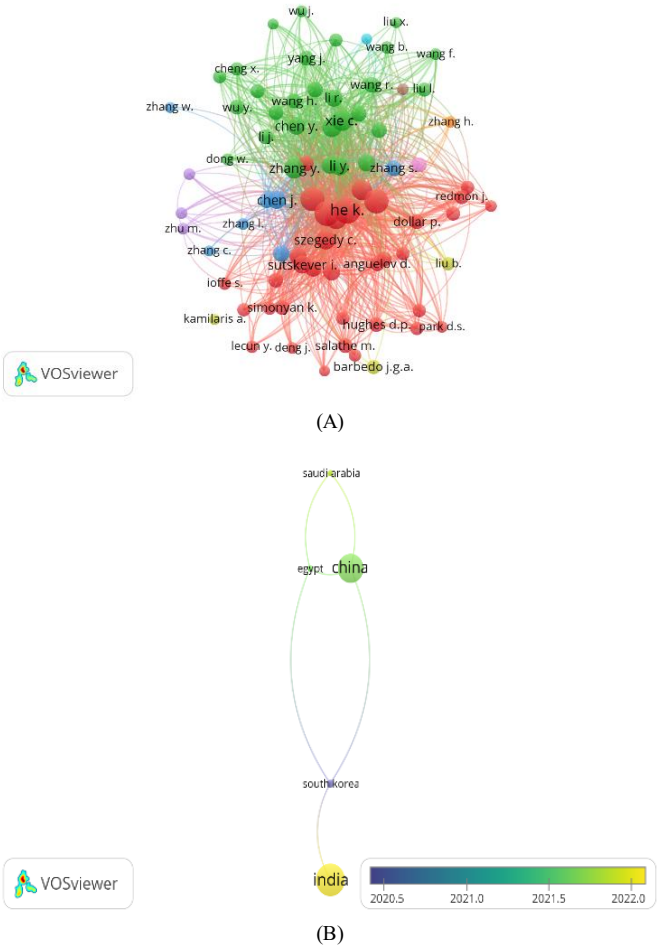


Fig. 7. Behavior of co-citation of articles by authors (7A) and countries (7B).

Fig. 7 shows the co-citation of the evaluated articles, this indicates a strong co-occurrence relationship that occurs when two items from the existing literature are cited [168], among them are the articles of Yang, Zhang, Wang, Chen and Xie. While in the country's co-citation is led by India and the most emerging citations is led by India and China. These results are related to the advancement of smart agriculture and have actively responded to climate change achieved sustainable breakthroughs [169–171]. This indicates that new technologies should be incorporated in all countries, especially in developing countries in order to close research gaps [172].

A. Comparison with Previous Research

The trends identified in this review are consistent with earlier bibliometric and application-oriented studies on artificial intelligence in agriculture. For example, recent analyses have also highlighted India and China as leading contributors to neural network-based pest detection, reflecting their rapid adoption of AI technologies in agriculture [26, 27, 169, 170]. Similar dominance of ResNet and YOLO architectures was reported in global surveys of deep learning for plant disease diagnosis, where these models achieved high accuracy across multiple crops [35, 39, 55, 70, 91]. However, our findings reveal an even stronger concentration of research in Asia than those reports, while Latin America and Africa remain underrepresented, echoing the gaps in agricultural automation noted in previous works [173–175]. These comparisons reinforce the need for international collaboration and targeted funding to expand AI-driven pest management research in developing regions.

IV. Conclusion

The studies of neural networks in the diagnosis of pests show that the countries of India and China lead in the publications, focused on corn and rice crops, the most used neural network architectures in the studies explored were ResNet and versions, YOLO and versions, VGG-16 showing a great performance, this is related to the keywords that stand out are "Convolution neural networks, image processing and Deep learning". In that sense, this review provides emerging research information, evidencing that there is a dire need to implement these advanced and innovative technologies, such as AI. By adopting these advanced technologies, farmers will improve their farming practices. The advancement of such advanced technologies will lead to sustainable agriculture.

Future research should also address the geographic imbalance identified in this review by promoting studies in underrepresented regions such as Latin America and Africa, and by conducting comparative evaluations of different neural network architectures under diverse climatic and cropping conditions to identify the models with the best cost-benefit ratio for real-world implementation. Likewise, collaborative initiatives among researchers, policy makers, and local farming communities would foster technology transfer and the practical adoption of AI-based pest management in smallholder and resource-limited agricultural systems.

Data Availability: The data used to support the findings of this study are available from the corresponding author upon request.

Funding: This research has been supported by the Faculty of Engineering and Systems and Electrical Mechanics, Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas (UNTRM).

Declaration of competing interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors acknowledge and thank the Institute for Research in Information and Communication Technologies (IITIC), Faculty of Systems Engineering, and Electrical Mechanics of the Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas (UNTRM).

References

[1] Sheema, D.; Ramesh, K.; Renjith, P.N.; Aiswarya, S. Fall Armyworm Detection on Maize Plants Using Gas Sensors, Image Classification, and Neural Network Based on IoT. International Journal of Intelligent Systems and Applications in En-gineering 2022, 10, 165–173.

[2] Poveda, J. Insect frass in the development of sustainable agriculture. A review. Agronomy for Sustainable Development 2021, 41, doi:10.1007/s13593-020-00656-x.

[3] Suman, J.; Rakshit, A.; Ogireddy, S.D.; Singh, S.; Gupta, C.; Chandrakala, J. Microbiome as a Key Player in Sustainable Agriculture and Human Health. Frontiers in Soil Science 2022, 2, 1–13, doi:10.3389/fsoil.2022.821589.

[4] Hartman, G.L.; West, E.D.; Herman, T.K. Crops that feed the World 2. Soybean-worldwide production, use, and constraints caused by pathogens and pests. Food Security 2011, 3, 5–17, doi:10.1007/s12571-010-0108-x.

[5] Vasantha, S.V.; Samreen, S.; Aparna, Y.L. Rice Disease Diagnosis System (RDDS). Computers, Materials and Continua 2022, 73, 1895–1914, doi:10.32604/cmc.2022.028504.

[6] Siddique, A.B.; Nasim, M.; Latif, A.; Dr, K.; Saiful, A.; Dr, I.; Iffekharuddaula, K.M.; Kashem, M.A.; Abdur, M.; Sarkar, R. A Way Forward to Combat Insect Pest in Rice. Bangladesh Rice 2020, 25.

[7] Espinoza Vanegas, W.L. Los cereales como fuente de alimentación primaria para la humanidad. Revista Multi-Ensayos 2018, 4, 47–54, doi:10.5377/multiensayos.v4i7.9493.

[8] Murdia L. K; Wadhawani, R.; Wadhawan, N.; Bajpai, P.; Shekhawat, S. Maize Utilization in India: An Overview. American Journal of Food and Nutrition 2016, 4, 169–176, doi:10.12691/ajfn-4-6-5.

[9] Babu, P.R.; Krishna, A.S. Deep Learning-Assisted SVMs for Efficacious Diagnosis of Tomato Leaf Diseases: A Comparative Study of GoogleNet, AlexNet, and ResNet-50. Ingenierie des Systemes d'Information 2023, 28, 639–645, doi:10.18280/isi.280312.

[10] 1Maican, E.; Iosif, A.; Maican, S. Precision Corn Pest Detection: Two-Step Transfer Learning for Beetles (Coleoptera) with MobileNet-SSD. Agriculture (Switzerland) 2023, 13, doi:10.3390/agriculture13122287.

[11] Food and Agriculture Organization of the United Nations (FAO) Producción y protección vegetal Acerca de la labor de la FAO en materia de producción y protección vegetal. 2024, 1–6.

[12] Loyani, L.; Machuve, D. A Deep Learning-based Mobile Application for Segmenting Tuta Absoluta's Damage on Tomato Plants. Engineering, Technology and Applied Science Research 2021, 11, 7730–7737, doi:10.48084/etasr.4355.

[13] Lima, M.C.F.; Leandro, M.E.D. de A.; Valero, C.; Coronel, L.C.P.; Bazzo, C.O.G. Automatic detection and monitoring of insect pests—A review. Agriculture (Switzerland) 2020, 10, doi:10.3390/agriculture10050161.

[14] Popescu, D.; Dinca, A.; Ichim, L.; Angelescu, N. New trends in detection of harmful insects and pests in modern agriculture using artificial neural networks. a review. Frontiers in Plant Science 2023, 14, 1–29, doi:10.3389/fpls.2023.1268167.

[15] De Cesaro Júnior, T.; Rieder, R. Automatic identification of insects from digital images: A survey. Computers and Elec-tronics in Agriculture 2020, 178, 105784, doi:10.1016/j.compag.2020.105784.

[16] Karami, A.; Crawford, M.; Delp, E.J. Automatic Plant Counting and Location Based on a Few-Shot Learning Technique. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 2020, 13, 5872–5886, doi:10.1109/JSTARS.2020.3025790.

[17] Esgario, J.G.M.; de Castro, P.B.C.; Tassis, L.M.; Krohling, R.A. An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning. Information Processing in Agriculture 2022, 9, 38–47, doi:10.1016/j.inpa.2021.01.004.

[18] Kavithamani, V.; UmaMaheswari, S. Investigation of deep learning for whitefly identification in coconut tree leaves. Intelligent Systems with Applications 2023, 20, 200290, doi:10.1016/j.iswa.2023.200290.

[19] Khanna, M.; Singh, L.K.; Thawkar, S.; Goyal, M. PlaNet: a robust deep convolutional neural network model for plant leaves disease recognition. Multimedia Tools and Applications 2024, 83, 4465–4517, doi:10.1007/s11042-023-15809-9.

[20] Feroza, A.Z.; Adiwijaya, N.O.; Putra, B.T.W. Development of a Web-based Application by Employing a Convolutional Neural Network (CNN) to Identify Pests and Diseases on Pakcoy (Brassica rapa subsp. chinensis). Pertanika Journal of Science and Technology 2023, 31, 2873–2885, doi:10.47836/pjst.31.6.13.

[21] Xiang, Q.; Huang, X.; Huang, Z.; Chen, X.; Cheng, J.; Tang, X. Yolo-Pest: An Insect Pest Object Detection Algorithm via CAC3 Module. Sensors 2023, 23, 1–16, doi:10.3390/s23063221.

[22] Khalid, S.; Oqaibi, H.M.; Aqib, M.; Hafeez, Y. Small Pests Detection in Field Crops Using Deep Learning Object Detection. Sustainability (Switzerland) 2023, 15, 1–19, doi:10.3390/su15086815.

[23] Newton, D.P. Quality and peer review of research: An adjudicating role for editors. Accountability in Research 2010, 17, 130–145, doi:10.1080/08989621003791945.

[24] Haddaway, N.R.; Bayliss, H.R. Shades of grey: Two forms of grey literature important for reviews in conservation. Bi-ological Conservation 2015, 191, 827–829, doi:10.1016/j.biocon.2015.08.018.

[25] van Eck, N.J.; Waltman, L. Software survey: VOSviewer, a computer program for bibliometric mapping. Scientometrics 2010, 84, 523–538, doi:10.1007/s11192-009-0146-3.

[26] Sabu, K.M.; Kumar, T.K.M. Predictive analytics in Agriculture: Forecasting prices of Arecanuts in Kerala. Procedia Computer Science 2020, 171, 699–708, doi:10.1016/j.procs.2020.04.076.

[27] Mahto, A.K.; Alam, M.A.; Biswas, R.; Ahmad, J.; Alam, S.I. Short-Term Forecasting of Agriculture Commodities in Context of Indian Market for Sustainable Agriculture by Using the Artificial Neural Network. Journal of Food Quality 2021, 2021, doi:10.1155/2021/9939906.

[28] Ndwiga, G.M. and J. This document is discoverable and free to researchers across the globe due to the work of AgEcon Search . Help ensure our sustainability. AgEcon Search 2015, 18.

[29] Niazian, M.; Niedbała, G. Machine learning for plant breeding and biotechnology. Agriculture (Switzerland) 2020, 10, 1–23, doi:10.3390/agriculture10100436.

[30] Liakos, K.G.; Busato, P.; Moshou, D.; Pearson, S.; Bochtis, D. Machine learning in agriculture: A review. Sensors (Swit-zerland) 2018, 18, 1–29, doi:10.3390/s18082674.

[31] Dias, P.A.; Tabb, A.; Medeiros, H. Apple flower detection using deep convolutional networks. Computers in Industry 2018, 99, 17–28, doi:10.1016/j.compind.2018.03.010.

[32] Khan, T.; Sherazi, H.H.R.; Ali, M.; Letchmunan, S.; Butt, U.M. Deep learning-based growth prediction system: A use case of china agriculture. Agronomy 2021, 11, 1–18, doi:10.3390/agronomy11081551.

[33] Nagaraj, G.; Sungeetha, D.; Tiwari, M.; Ahuja, V.; Varma, A.K.; Agarwal, P. Advancements in Plant Pests Detection: Leveraging Convolutional Neural Networks for Smart Agriculture. 2024, 221, doi:10.3390/engproc2023059201.

[34] Robert, M.; Dury, J.; Thomas, A.; Therond, O.; Sekhar, M.; Badiger, S.; Ruiz, L.; Bergez, J.E. CMFDM: A methodology to guide the design of a conceptual model of farmers' decision-making processes. Agricultural Systems 2016, 148, 86–94, doi:10.1016/j.agsy.2016.07.010.

[35] Yang, L.; Yu, X.; Zhang, S.; Long, H.; Zhang, H.; Xu, S.; Liao, Y. GoogLeNet based on residual network and attention mechanism identification of rice leaf diseases. Computers and Electronics in Agriculture 2023, 204, 107543, doi:10.1016/j.compag.2022.107543.

[36] Attri, I.; Awasthi, L.K.; Sharma, T.P.; Rathee, P. A review of deep learning techniques used in agriculture. Ecological In-formatics 2023, 77, 102217, doi:10.1016/j.ecoinf.2023.102217.

[37] Silva, M.F.; LuÃ-s Lima, J.; Reis, L.P.; Sanfeliu, A.; Tardioli, D. Correction: Robot 2019: Fourth Iberian Robotics Conference (Adv. Intell. Sys. Comput. (2019), 1092 AISC, 10.1007/978-3-030-35990-4_55); 2020; Vol 1092 AISC; ISBN 9783030359898.

[38] Wang, C.; Liu, B.; Liu, L.; Zhu, Y.; Hou, J.; Liu, P.; Li, X. A review of deep learning used in the hyperspectral image analysis for agriculture; Springer Netherlands, 2021; Vol 54; ISBN 0123456789.

[39] Zhang, X.; Qiao, Y.; Meng, F.; Fan, C.; Zhang, M. Identification of maize leaf diseases using improved deep convolutional neural networks. IEEE Access 2018, 6, 30370–30377, doi:10.1109/ACCESS.2018.2844405.

[40] About, L.; Help, B. Corn leaf disease recognition based on support vector machine method. 2007, 3–4.

[41] Bergougnoux, V. The history of tomato: From domestication to biopharming. Biotechnology Advances 2014, 32, 170–189, doi:10.1016/j.biotechadv.2013.11.003.

[42] Polin, J.A.; Hasan, N.; Habib, M.T.; Rahman, A.; Vasha, Z.N.; Sharma, B. Tomato pest recognition using convolutional neural network in Bangladesh. Bulletin of Electrical Engineering and Informatics 2024, 13, 619–627, doi:10.11591/eei.v13i1.6073.

[43] Huang, M.L.; Chuang, T.C.; Liao, Y.C. Application of transfer learning and image augmentation technology for tomato pest identification. Sustainable Computing: Informatics and Systems 2022, 33, 100646, doi:10.1016/j.suscom.2021.100646.

[44] Ngugi, L.C.; Abelwahab, M.; Abo-Zahhad, M. Recent advances in image processing techniques for automated leaf pest and disease recognition – A review. Information Processing in Agriculture 2021, 8, 27–51, doi:10.1016/j.inpa.2020.04.004.

[45] Babaee, M.; Maroufpoor, S.; Jalali, M.; Zarei, M.; Elbeltagi, A. Artificial intelligence approach to estimating rice yield*. Irrigation and Drainage 2021, 70, 732–742, doi:10.1002/ird.2566.

[46] Acevedo, M.A.; Castrillo, W.A. ORIGEN, EVOLUCIÓN Y DIVERSIDAD DEL ARROZ. Agronomía Tropical 2006, 56, 1–10.

[47] Jiang, Y.; Carrijo, D.; Huang, S.; Chen, J.; Balaine, N.; Zhang, W.; van Groenigen, K.J.; Linquist, B. Water management to mitigate the global warming potential of rice systems: A global meta-analysis. Field Crops Research 2019, 234, 47–54, doi:10.1016/j.fcr.2019.02.010.

[48] Ramos-Fernández, L.; Quispe-Tito, D.; Altamirano-Gutiérrez, L.; Cruz-Grimaldo, C.; Quille-Mamani, J.A.; Car-bonell-Rivera, J.P.; Torralba, J.; Ruiz, L.Á. Estimation of evapotranspiration from UAV high-resolution images for irri-gation systems in rice fields on the northern coast of Peru. Scientia Agropecuaria 2024, 15, 7–21, doi:10.17268/sci.agropecu.2024.001.

[49] Ilham Rahmana Syihad; Muhammad Rizal; Zamah Sari; Yufis Azhar CNN Method to Identify the Banana Plant Diseases based on Banana Leaf Images by Giving Models of ResNet50 and VGG-19. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi) 2023, 7, 1309–1318, doi:10.29207/resti.v7i6.5000.

[50] Samridhi, S.; Kalpana, M.; Parimalarangan, R.; Palanichamy, N.V. Identification of Sigatoka Leaf Spot Disease in Banana Using Convolutional Neural Network (CNN). Asian Journal of Agricultural Extension, Economics & Sociology 2023, 41, 931–936, doi:10.9734/ajaees/2023/v41i92123.

[51] Kumar, A.; Patel, V.K. Classification and identification of disease in potato leaf using hierarchical based deep learning convolutional neural network. Multimedia Tools and Applications 2023, 82, 31101–31127, doi:10.1007/s11042-023-14663-z.

[52] Sachdeva, G.; Singh, P.; Kaur, P. Plant leaf disease classification using deep Convolutional neural network with Bayesian learning. Materials Today: Proceedings 2021, 45, 5584–5590, doi:10.1016/j.matpr.2021.02.312.

[53] Lin, J.; Chen, X.; Pan, R.; Cao, T.; Cai, J.; Chen, Y.; Peng, X.; Cernava, T.; Zhang, X. GrapeNet: A Lightweight Convolutional Neural Network Model for Identification of Grape Leaf Diseases. Agriculture (Switzerland) 2022, 12, doi:10.3390/agriculture12060887.

[54] Nandhini, C.; Brindha, M. Visual regenerative fusion network for pest recognition. Neural Computing and Applications 2024, 36, 2867–2882, doi:10.1007/s00521-023-09173-w.

[55] Tang, Z.; Lu, J.; Chen, Z.; Qi, F.; Zhang, L. Improved Pest-YOLO: Real-time pest detection based on efficient channel at-tention mechanism and transformer encoder. Ecological Informatics 2023, 78, 102340, doi:10.1016/j.ecoinf.2023.102340.

[56] Vedhamuru, N.; Malmathanraj, R.; Palanisamy, P. Features of pyramid dilation rate with residual connected convolution neural network for pest classification. Signal, Image and Video Processing 2024, 18, 715–722, doi:10.1007/s11760-023-02712-x.

[57] Radha, M.D.; Prasanna, S. Tomato Crop Yield Prediction in Indoor Environment with A Novel ABC Enhanced CNN with SDL Architecture. International Journal of Intelligent Systems and Applications in Engineering 2024, 12, 217–232.

[58] Lin, J.; Chen, X.; Cai, J.; Pan, R.; Cernava, T.; Migheli, Q.; Zhang, X.; Qin, Y. Looking from shallow to deep: Hierarchical complementary networks for large scale pest identification. Computers and Electronics in Agriculture 2023, 214, 108342, doi:10.1016/j.compag.2023.108342.

[59] Gupta, V.A.; Padmavati, M. V.; Saxena, R.R. a Novel Approach for Insect-Pest Identification Using Multipath Convolu-tional Neural Network. Agricultural Research Journal 2023, 60, 706–711, doi:10.5958/2395-146X.2023.00100.X.

[60] Hua, J.; Zhu, T.; Zou, F.; Zou, J.; Tang, J. CSLSNet: A Compressed Domain Classification Model for Pest and Disease Images. Agronomy 2023, 13, 1–24, doi:10.3390/agronomy13102663.

[61] Singh, P.; Singh, P.; Farooq, U.; Khurana, S.S.; Verma, J.K.; Kumar, M. CottonLeafNet: cotton plant leaf disease detection using deep neural networks. Multimedia Tools and Applications 2023, 82, 37151–37176, doi:10.1007/s11042-023-14954-5.

[62] Dai, G.; Fan, J.; Dewi, C. ITF-WPI: Image and text based cross-modal feature fusion model for wolfberry pest recognition. Computers and Electronics in Agriculture 2023, 212, 108129, doi:10.1016/j.compag.2023.108129.

[63] Math, R.K.M.; Dharwadkar, N. V. Deep learning and computer vision for leaf miner infestation severity detection on muskmelon (Cucumis melo) leaves. Computers and Electrical Engineering 2023, 110, 108843, doi:10.1016/j.compeleceng.2023.108843.

[64] Raja, N.B.; Rajendran, P.S. A Novel Fuzzy-Based Modified GAN and Faster RCNN for Classification of Banana Leaf Disease. Journal of The Institution of Engineers (India): Series A 2023, 104, 529–540, doi:10.1007/s40030-023-00743-8.

[65] Shafik, W.; Tufail, A.; Liyanage, C.D.S.; Anna, R.; Haji, A.; Apong, M. Using a novel convolutional neural network for plant pests detection and disease classification. Journal of the Science of Food and Agriculture 2023, 1–9.

[66] Kasinathan, T.; Uyyala, S.R. Detection of fall armyworm (spodoptera frugiperda) in field crops based on mask R-CNN. Signal, Image and Video Processing 2023, 17, 2689–2695, doi:10.1007/s11760-023-02485-3.

[67] Sanghavi, V.B.; Bhadka, H.; Dubey, V. Hunger games search based deep convolutional neural network for crop pest identification and classification with transfer learning. Evolving Systems 2023, 14, 649–671, doi:10.1007/s12530-022-09449-x.

[68] Chand, L.; Dhiman, A.S.; Singh, S. Detection of whitefly pests in crops employing image enhancement and machine learning. International Journal of Advanced Technology and Engineering Exploration 2023, 10, 569–589, doi:10.19101/IJATEE.2022.10100289.

[69] Xu, Y. Research on lightweight target detection algorithm of farmland insect pests based on YOLO-PPLCBot. Journal of Electronic Imaging 2023, 32, 1–2.

[70] Liu, D.; Lv, F.; Guo, J.; Zhang, H.; Zhu, L. Detection of Forestry Pests Based on Improved YOLOv5 and Transfer Learning. Forests 2023, 14, doi:10.3390/f14071484.

[71] Devi, N.; Sarma, K.K.; Laskar, S. Design of an intelligent bean cultivation approach using computer vision, IoT and spa-tio-temporal deep learning structures. Ecological Informatics 2023, 75, 102044, doi:10.1016/j.ecoinf.2023.102044.

[72] Atheeswaran, A.; Raghavender, K. V.; Chaganti, B.N.L.; Maram, A.; Herencsar, N. Expert system for smart farming for diagnosis of sugarcane diseases using machine learning. Computers and Electrical Engineering 2023, 109, 108739, doi:10.1016/j.compeleceng.2023.108739.

[73] Bocca, P.; Orellana, A.; Soria, C.; Carelli, R. On field disease detection in olive tree with vision systems. Array 2023, 18, 100286, doi:10.1016/j.array.2023.100286.

[74] Fallah, M.; Parmehr, E.G. Detection of Chilo Suppressalis using Smartphone Images and Deep Learning. Journal of Ag-ricultural Machinery 2023, 13, 195–211, doi:10.22067/jam.2022.72647.1064.

[75] Wang, Y.; Chen, Y.; Wang, D. Convolution Network Enlightened Transformer for Regional Crop Disease Classification. Electronics 2022, 11.

[76] Parasa, G.; Arulselvi, M.; Razia, S. Identification of Diseases in Paddy Crops Using CNN. International Journal of Intel-ligent Systems and Applications in Engineering 2023, 11, 548–557.

[77] Chand, L.; Dhiman, A.S.; Singh, S. A multi-instance learning based approach for whitefly pest detection. Indonesian Journal of Electrical Engineering and Computer Science 2023, 31, 1050–1060, doi:10.11591/ijeecs.v31.i2.pp1050-1060.

[78] Čirjak, D.; Aleksi, I.; Lemic, D.; Pajač Živković, I. EfficientDet-4 Deep Neural Network-Based Remote Monitoring of Codling Moth Population for Early Damage Detection in Apple Orchard. Agriculture (Switzerland) 2023, 13, doi:10.3390/agriculture13050961.

[79] Prasath, B.; Akila, M. IoT-based pest detection and classification using deep features with enhanced deep learning strategies. Engineering Applications of Artificial Intelligence 2023, 121, 105985, doi:10.1016/j.engappai.2023.105985.

[80] Lin, S.; Xiu, Y.; Kong, J.; Yang, C.; Zhao, C. An Effective Pyramid Neural Network Based on Graph-Related Attentions Structure for Fine-Grained Disease and Pest Identification in Intelligent Agriculture. Agriculture (Switzerland) 2023, 13, doi:10.3390/agriculture13030567.

[81] Obasekore, H.; Fanni, M.; Ahmed, S.M.; Parque, V.; Kang, B.Y. Agricultural Robot-Centered Recognition of Ear-ly-Developmental Pest Stage Based on Deep Learning: A Case Study on Fall Armyworm (Spodoptera frugiperda). Sensors 2023, 23, doi:10.3390/s23063147.

[82] Arshaghi, A.; Ashourian, M.; Ghabeli, L. Potato diseases detection and classification using deep learning methods. Mul-timedia Tools and Applications 2023, 82, 5725–5742, doi:10.1007/s11042-022-13390-1.

[83] Gan, S.; Zhou, D.; Cui, Y.; Lv, J. A Neural Network-based Approach for Apple Leaf Disease Detection in Smart Agriculture Application. International Journal of Advanced Computer Science and Applications 2023, 14, 568–573, doi:10.14569/IJACSA.2023.0141158.

[84] Chen, Y.; Chen, M.; Guo, M.; Wang, J.; Zheng, N. Pest recognition based on multi-image feature localization and adaptive filtering fusion. Frontiers in Plant Science 2023, 14, 1–14, doi:10.3389/fpls.2023.1282212.

[85] Sushma Sri, V.; Hima Sailu, V.; Pradeepthi, U.; Manogyna Sai, P.; Kavitha, M. Disease Detection using Region-Based Convolutional Neural Network and ResNet. Data and Metadata 2023, 2, doi:10.56294/dm2023135.

[86] Raja, D.; Karthikeyan, M. Red Deer Optimization with Deep Learning Enabled Agricultural Plant Disease Detection and Classification Model. International Journal of Intelligent Engineering and Systems 2023, 16, 21–30, doi:10.22266/ijies2023.1031.03.

[87] Guan, H.; Fu, C.; Zhang, G.; Li, K.; Wang, P.; Zhu, Z. A lightweight model for efficient identification of plant diseases and pests based on deep learning. Frontiers in Plant Science 2023, 14, 1–13, doi:10.3389/fpls.2023.1227011.

[88] Rathi, S.; Nirgude, V. Improving the Accuracy of Real Field Pomegranate Fruit Diseases Detection and Visualization using Convolution Neural Networks and Grad-CAM. International Journal of Data Analysis Techniques and Strategies 2023, 15, 1, doi:10.1504/ijdats.2023.10055387.

[89] Sardar, P.; Ema, R.R.; Kabir, S.S.; Adnan, N.; Galib, S.M. Severity Stage Identification and Pest Detection of Tomato Disease Using Deep Learning. International Journal of Computing 2023, 22, 191–201, doi:10.47839/ijc.22.2.3088.

[90] Yao, J.; Liu, J.; Zhang, Y.; Wang, H. Identification of winter wheat pests and diseases based on improved convolutional neural network. Open Life Sciences 2023, 18, 1–11, doi:10.1515/biol-2022-0632.

[91] Zhang, C.; Hu, Z.; Xu, L.; Zhao, Y. A YOLOv7 incorporating the Adan optimizer based corn pests identification method. Frontiers in Plant Science 2023, 14, 1–14, doi:10.3389/fpls.2023.1174556.

[92] Gupta, V.A.; Padmavati, M. V.; Saxena, R.R.; Tamrakar, R.K. A Novel Insect and Pest Identification Model Based on a Weighted Multipath Convolutional Neural Network and Generative Adversarial Network. Karbala International Journal of Modern Science 2023, 9, 149–159, doi:10.33640/2405-609X.3280.

[93] Čirjak, D.; Aleksi, I.; Miklečić, I.; Antolković, A.M.; Vrtodušić, R.; Viduka, A.; Lemic, D.; Kos, T.; Pajač Živković, I. Mon-itoring System for Leucoptera malifoliella (O. Costa, 1836) and Its Damage Based on Artificial Neural Networks. Agri-culture (Switzerland) 2023, 13, doi:10.3390/agriculture13010067.

[94] Aladhadh, S.; Habib, S.; Islam, M.; Aloraini, M.; Aladhadh, M.; Al-Rawashdeh, H.S. An Efficient Pest Detection Framework with a Medium-Scale Benchmark to Increase the Agricultural Productivity. Sensors 2022, 22, 1–17, doi:10.3390/s22249749.

[95] Çetiner, H. Citrus disease detection and classification using based on convolution deep neural network. Microprocessors and Microsystems 2022, 95, 104687, doi:10.1016/j.micpro.2022.104687.

[96] Momeny, M.; Jahanbakhshi, A.; Neshat, A.A.; Hadipour-Rokni, R.; Zhang, Y.D.; Ampatzidis, Y. Detection of citrus black spot disease and ripeness level in orange fruit using learning-to-augment incorporated deep networks. Ecological In-formatics 2022, 71, 101829, doi:10.1016/j.ecoinf.2022.101829.

[97] Cheng, H.H.; Dai, Y.L.; Lin, Y.; Hsu, H.C.; Lin, C.P.; Huang, J.H.; Chen, S.F.; Kuo, Y.F. Identifying tomato leaf diseases under real field conditions using convolutional neural networks and a chatbot. Computers and Electronics in Agriculture 2022, 202, 107365, doi:10.1016/j.compag.2022.107365.

[98] Dey, B.; Masum Ul Haque, M.; Khatun, R.; Ahmed, R. Comparative performance of four CNN-based deep learning var-iants in detecting Hispa pest, two fungal diseases, and NPK deficiency symptoms of rice (Oryza sativa). Computers and Electronics in Agriculture 2022, 202, 107340, doi:10.1016/j.compag.2022.107340.

[99] Coulibaly, S.; Kamsu-Foguem, B.; Kamissoko, D.; Traore, D. Explainable deep convolutional neural networks for insect pest recognition. Journal of Cleaner Production 2022, 371, 133638, doi:10.1016/j.jclepro.2022.133638.

[100] Moussafir, M.; Chaibi, H.; Saadane, R.; Chehri, A.; Rharras, A. El; Jeon, G. Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks. Plant and Soil 2022, 479, 251–266, doi:10.1007/s11104-022-05513-2.

[101] Yang, C.; Teng, Z.; Dong, C.; Lin, Y.; Chen, R.; Wang, J. In-Field Citrus Disease Classification via Convolutional Neural Network from Smartphone Images. Agriculture (Switzerland) 2022, 12, 1–11, doi:10.3390/agriculture12091487.

[102] Setiawan, A.; Yudistira, N.; Wihandika, R.C. Large scale pest classification using efficient Convolutional Neural Network with augmentation and regularizers. Computers and Electronics in Agriculture 2022, 200, 107204, doi:10.1016/j.compag.2022.107204.

[103] Li, W.; Zhu, T.; Li, X.; Dong, J.; Liu, J. Recommending Advanced Deep Learning Models for Efficient Insect Pest Detection. Agriculture (Switzerland) 2022, 12, 1–17, doi:10.3390/agriculture12071065.

[104] Math, R.K.M.; Dharwadkar, N. V. Early detection and identification of grape diseases using convolutional neural net-works. Journal of Plant Diseases and Protection 2022, 129, 521–532, doi:10.1007/s41348-022-00589-5.

[105] Chodey, M.D.; Noorullah Shariff, C. Hybrid deep learning model for in-field pest detection on real-time field monitoring. Journal of Plant Diseases and Protection 2022, 129, 635–650, doi:10.1007/s41348-022-00584-w.

[106] Qiang, Z.; Shi, F. Pest disease detection of Brassica chinensis in wide scenes via machine vision: method and deployment. Journal of Plant Diseases and Protection 2022, 129, 533–544, doi:10.1007/s41348-021-00562-8.

[107] Anitha, J.; Saranya, N. Cassava Leaf Disease Identification and Detection Using Deep Learning Approach. International Journal of Computers, Communications and Control 2022, 17, doi:10.15837/ijccc.2022.2.4356.

[108] Feng, J.; Sun, Y.; Zhang, K.; Zhao, Y.; Ren, Y.; Chen, Y.; Zhuang, H.; Chen, S. Autonomous Detection of Spodoptera fru-giperda by Feeding Symptoms Directly from UAV RGB Imagery. Applied Sciences (Switzerland) 2022, 12, doi:10.3390/app12052592.

[109] Turkoglu, M.; Yanikoğlu, B.; Hanbay, D. PlantDiseaseNet: convolutional neural network ensemble for plant disease and pest detection. Signal, Image and Video Processing 2022, 16, 301–309, doi:10.1007/s11760-021-01909-2.

[110] Butera, L.; Ferrante, A.; Jermini, M.; Prevostini, M.; Alippi, C. Precise Agriculture: Effective Deep Learning Strategies to Detect Pest Insects. IEEE/CAA Journal of Automatica Sinica 2022, 9, 246–258, doi:10.1109/JAS.2021.1004317.

[111] Omer, S.M.; Ghafoor, K.Z.; Askar, S.K. An Intelligent System for Cucumber Leaf Disease Diagnosis Based on the Tuned Convolutional Neural Network Algorithm. Mobile Information Systems 2022, 2022, doi:10.1155/2022/8909121.

[112] Alagesan, M.; Kesavan, T.; Murugesan, H.; Thangavel, M.; Madesh, G. Plant disease detection. AIP Conference Proceedings 2023, 2857, 321–333, doi:10.1063/5.0164983.

[113] Nader, A.; Khafagy, M.H.; Hussien, S.A. Grape Leaves Diseases Classification using Ensemble Learning and Transfer Learning. International Journal of Advanced Computer Science and Applications 2022, 13, 563–571, doi:10.14569/IJACSA.2022.0130767.

[114]Ullah, N.; Khan, J.A.; Alharbi, L.A.; Raza, A.; Khan, W.; Ahmad, I. An Efficient Approach for Crops Pests Recognition and Classification Based on Novel DeepPestNet Deep Learning Model. IEEE Access 2022, 10, 73019–73032, doi:10.1109/ACCESS.2022.3189676.

[115]Zhang, S.; Jing, R.; Shi, X. Crop pest recognition based on a modified capsule network. Systems Science and Control Engineering 2022, 10, 552–561, doi:10.1080/21642583.2022.2074168.

[116]Gan, Y.; Guo, Q.; Wang, C.; Liang, W.; Xiao, D.; Wu, H. Recognizing crop pests using an improved EfficientNet model. Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering 2022, 38, 203–211, doi:10.11975/j.issn.1002-6819.2022.01.023.

[117]Nirgude, V.; Rathi, S. a Robust Deep Learning Approach To Enhance the Accuracy of Pomegranate Fruit Disease Detection Under Real Field Condition. Journal of Experimental Biology and Agricultural Sciences 2021, 9, 863–870, doi:10.18006/2021.9(6).863.870.

[118]Takkar, S.; Kakran, A.; Kaur, V.; Rakhra, M.; Sharma, M.; Bangotra, P.; Verma, N. Recognition of Image-Based Plant Leaf Diseases Using Deep Learning Classification Models. Nature Environment and Pollution Technology 2021, 20, 2137–2147, doi:10.46488/NEPT.2021.V20I05.031.

[119]Sun, J.; Zhu, W.; Luo, Y.; Shen, J.; Chen, Y.; Zhou, X. Recognizing the diseases of crop leaves in fields using improved Mobilenet-V2. Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering 2021, 37, 161–169, doi:10.11975/j.issn.1002-6819.2021.22.018.

[120]Grünig, M. Applying deep neural networks to predict incidence and phenology of plant pests and diseases. Emerging Technologies 2021, 12, 105–128, doi:10.1201/9781482294514-8.

[121]Afzaal, U.; Bhattarai, B.; Pandeya, Y.R.; Lee, J. An instance segmentation model for strawberry diseases based on mask R-CNN. Sensors 2021, 21, doi:10.3390/s21196565.

[122]Bao, W.; Yang, X.; Liang, D.; Hu, G.; Yang, X. Lightweight convolutional neural network model for field wheat ear disease identification. Computers and Electronics in Agriculture 2021, 189, 106367, doi:10.1016/j.compag.2021.106367.

[123]Karar, M.E.; Alsunaydi, F.; Albusaymi, S.; Alotaibi, S. A new mobile application of agricultural pests recognition using deep learning in cloud computing system. Alexandria Engineering Journal 2021, 60, 4423–4432, doi:10.1016/j.aej.2021.03.009.

[124]Zhang, S.; Xu, X.; Qi, G.; Shao, Y. Detecting the pest disease of field crops using deformable VGG-16 model. Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering 2021, 37, 188–194, doi:10.11975/j.issn.1002-6819.2021.18.022.

[125]Hadipour Rokni, R.; Askari Asli-Ardeh, E.; Sabzi, S.; Esmaili Paeen-Afrakoti, I. Detection of snail pest in citrus orchard under different lighting conditions using deep neural networks. Journal of Food Science and Technology (Iran) 2021, 18, 157–169, doi:10.29252/fsct.18.06.12.

[126]N, K.; Narasimha Prasad, L. V.; Pavan Kumar, C.S.; Subedi, B.; Abraha, H.B.; Sathishkumar, V.E. Rice leaf diseases pre-diction using deep neural networks with transfer learning. Environmental Research 2021, 198, 111275, doi:10.1016/j.envres.2021.111275.

[127]Muppala, C.; Guruviah, V. Detection of leaf folder and yellow stemborer moths in the paddy field using deep neural network with search and rescue optimization. Information Processing in Agriculture 2021, 8, 350–358, doi:10.1016/j.inpa.2020.09.002.

[128]Caldeira, R.F.; Santiago, W.E.; Teruel, B. Identification of cotton leaf lesions using deep learning techniques. Sensors 2021, 21, doi:10.3390/s21093169.

[129]Rustia, D.J.A.; Chao, J.J.; Chiu, L.Y.; Wu, Y.F.; Chung, J.Y.; Hsu, J.C.; Lin, T. Te Automatic greenhouse insect pest detection and recognition based on a cascaded deep learning classification method. Journal of Applied Entomology 2021, 145, 206–222, doi:10.1111/jen.12834.

[130]Gutiérrez, S.; Hernández, I.; Ceballos, S.; Barrio, I.; Díez-Navajas, A.M.; Tardaguila, J. Deep learning for the differentiation of downy mildew and spider mite in grapevine under field conditions. Computers and Electronics in Agriculture 2021, 182, 1–9, doi:10.1016/j.compag.2021.105991.

[131]Singh, P.; Verma, A.; Alex, J.S.R. Disease and pest infection detection in coconut tree through deep learning techniques. Computers and Electronics in Agriculture 2021, 182, 105986, doi:10.1016/j.compag.2021.105986.

[132]Muppala, C.; Guruviah, V. Paddy Pest Identification with Deep Convolutional Neural Networks. Engineering in Agri-culture, Environment and Food 2021, 14, 54–60, doi:10.37221/eaef.14.2_54.

[133]Urmashev, B.; Buribayev, Z.; Amirgaliyeva, Z.; Ataniyazova, A.; Zhassuzak, M.; Turegali, A. Development of a Weed Detection System Using Machine Learning and Neural Network Algorithms. Eastern-European Journal of Enterprise Technologies 2021, 6, 70–85, doi:10.15587/1729-4061.2021.246706.

[134]Jabir, B.; Rabhi, L.; Falih, N. RNN- and CNN-based weed detection for crop improvement: An overview. Foods and Raw Materials 2021, 9, 387–396, doi:10.21603/2308-4057-2021-2-387-396.

[135]Pattnaik, G.; Shrivastava, V.K.; Parvathi, K. Tomato pest classification using deep convolutional neural network with transfer learning, fine tuning and scratch learning. Intelligent Decision Technologies 2021, 15, 433–442, doi:10.3233/IDT-200192.

[136]Alosaimi, W.; Alyami, H.; Uddin, M.I. PeachNet: Peach diseases detection for automatic harvesting. Computers, Materials and Continua 2021, 67, 1665–1677, doi:10.32604/cmc.2021.014950.

[137]Malathi, V.; Gopinath, M.P. Classification of pest detection in paddy crop based on transfer learning approach. Acta Agriculturae Scandinavica Section B: Soil and Plant Science 2021, 71, 552–559, doi:10.1080/09064710.2021.1874045.

[138]Chen, C.J.; Huang, Y.Y.; Li, Y.S.; Chen, Y.C.; Chang, C.Y.; Huang, Y.M. Identification of Fruit Tree Pests with Deep Learning on Embedded Drone to Achieve Accurate Pesticide Spraying. IEEE Access 2021, 9, 21986–21997, doi:10.1109/ACCESS.2021.3056082.

[139]Bhagwat, R.; Dandawate, Y. Comprehensive Multilayer Convolutional Neural Network for Plant Disease Detection. International Journal of Advanced Computer Science and Applications 2021, 12, 204–211, doi:10.14569/IJACSA.2021.0120125.

[140]Lee, S.H.; Goëau, H.; Bonnet, P.; Joly, A. Attention-Based Recurrent Neural Network for Plant Disease Classification. Frontiers in Plant Science 2020, 11, 1–8, doi:10.3389/fpls.2020.601250.

[141]Wang, J.; Li, Y.; Feng, H.; Ren, L.; Du, X.; Wu, J. Common pests image recognition based on deep convolutional neural network. Computers and Electronics in Agriculture 2020, 179, 105834, doi:10.1016/j.compag.2020.105834.

[142]Ramalingam, B.; Mohan, R.E.; Pookkuttath, S.; Gómez, B.F.; Sairam Borusu, C.S.C.; Teng, T.W.; Tamilselvam, Y.K. Remote insects trap monitoring system using deep learning framework and iot. Sensors (Switzerland) 2020, 20, 1–17, doi:10.3390/s20185280.

[143]Xing, S.; Lee, M. Classification accuracy improvement for small-size citrus pests and diseases using bridge connections in deep neural networks. Sensors (Switzerland) 2020, 20, 1–16, doi:10.3390/s20174992.

[144]Roosjen, P.P.J.; Kellenberger, B.; Kooistra, L.; Green, D.R.; Fahrentrapp, J. Deep learning for automated detection of Drosophila suzukii: potential for UAV-based monitoring. Pest Management Science 2020, 76, 2994–3002, doi:10.1002/ps.5845.

[145]Shao, Z.Z.; Yao, Q.; Tang, J.; Li, H.Q.; Yang, B.J.; Lü, J.; Chen, Y. Research and development of the intelligent identification system of agricultural pests for mobile terminals. Scientia Agricultura Sinica 2020, 53, 3257–3268, doi:10.3864/j.issn.0578-1752.2020.16.005.

[146]Alves, A.N.; Souza, W.S.R.; Borges, D.L. Cotton pests classification in field-based images using deep residual networks. Computers and Electronics in Agriculture 2020, 174, 105488, doi:10.1016/j.compag.2020.105488.

[147]Rocha, A. V.; Lagarteja, J.G. Philippine carabao mango pest identification using convolutional neural network. Interna-tional Journal of Scientific and Technology Research 2020, 9, 3443–3448.

[148]Esgario, J.G.M.; Krohling, R.A.; Ventura, J.A. Deep learning for classification and severity estimation of coffee leaf biotic stress. Computers and Electronics in Agriculture 2020, 169, doi:10.1016/j.compag.2019.105162.

[149]Khalifa, N.E.M.; Loey, M.; Taha, M.H.N. Insect pests recognition based on deep transfer learning models. Journal of Theoretical and Applied Information Technology 2020, 98, 60–68.

[150]Pattnaik, G.; Shrivastava, V.K.; Parvathi, K. Transfer Learning-Based Framework for Classification of Pest in Tomato Plants. Applied Artificial Intelligence 2020, 34, 981–993, doi:10.1080/08839514.2020.1792034.

[151]Lin, T.L.; Chang, H.Y.; Chen, K.H. The pest and disease identification in the growth of sweet peppers using faster r-cnN and mask r-CNn. Journal of Internet Technology 2020, 21, 605–614, doi:10.3966/160792642020032102027.

[152]Yang, H.; Gao, L.; Tang, N.; Yang, P. Experimental analysis and evaluation of wide residual networks based agricultural disease identification in smart agriculture system. Eurasip Journal on Wireless Communications and Networking 2019, 2019, doi:10.1186/s13638-019-1613-z.

[153]Picon, A.; Alvarez-Gila, A.; Seitz, M.; Ortiz-Barredo, A.; Echazarra, J.; Johannes, A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. Computers and Electronics in Agriculture 2019, 161, 280–290, doi:10.1016/j.compag.2018.04.002.

[154]Li, R.; Jia, X.; Hu, M.; Zhou, M.; Li, D.; Liu, W.; Wang, R.; Zhang, J.; Xie, C.; Liu, L.; et al. An Effective Data Augmentation Strategy for CNN-Based Pest Localization and Recognition in the Field. IEEE Access 2019, 7, 160274–160283, doi:10.1109/ACCESS.2019.2949852.

[155]Thenmozhi, K.; Srinivasulu Reddy, U. Crop pest classification based on deep convolutional neural network and transfer learning. Computers and Electronics in Agriculture 2019, 164, 104906, doi:10.1016/j.compag.2019.104906.

[156]Dawei, W.; Limiao, D.; Jiangong, N.; Jiyue, G.; Hongfei, Z.; Zhongzhi, H. Recognition pest by image-based transfer learning. Journal of the Science of Food and Agriculture 2019, 99, 4524–4531, doi:10.1002/jsfa.9689.

[157]Xing, S.; Lee, M.; Lee, K.K. Citrus pests and diseases recognition model using weakly dense connected convolution network. Sensors (Switzerland) 2019, 19, doi:10.3390/s19143195.

[158]Picon, A.; Seitz, M.; Alvarez-Gila, A.; Mohnke, P.; Ortiz-Barredo, A.; Echazarra, J. Crop conditional Convolutional Neural Networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field con-ditions. Computers and Electronics in Agriculture 2019, 167, 105093, doi:10.1016/j.compag.2019.105093.

[159]We, E. Deep learning-based real-time monitoring and early warning system for grain storage pests. Journal of Jiangsu University 2019, 40, 2–3, doi:10.3969/j.issn.1671-7775.2019.02.013.

[160]Li, R.; Wang, R.; Xie, C.; Liu, L.; Zhang, J.; Wang, F.; Liu, W. A coarse-to-fine network for aphid recognition and detection in the field. Biosystems Engineering 2019, 187, 39–52, doi:10.1016/j.biosystemseng.2019.08.013.

[161]Mishra, M.; Singh, P.K.; Brahmachari, A.; Debnath, N.C.; Choudhury, P. A robust pest identification system using mor-phological analysis in neural networks. Projections (New York) 2019, 7, 483–495, doi:10.3167/PROJ.2019.130303.

[162]Da Silva, L.A.; Bressan, P.O.; Gonçalves, D.N.; Freitas, D.M.; Machado, B.B.; Gonçalves, W.N. Estimating soybean leaf defoliation using convolutional neural networks and synthetic images. Computers and Electronics in Agriculture 2019, 156, 360–368, doi:10.1016/j.compag.2018.11.040.

[163]Nazri, A.; Mazlan, N.; Muharam, F. Research article Penyek: Automated brown planthopper detection from imperfect sticky pad images using deep convolutional neural network. PLoS ONE 2018, 13, 1–13, doi:10.1371/journal.pone.0208501.

[164]Fuentes, A.; Yoon, S.; Kim, S.C.; Park, D.S. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors (Switzerland) 2017, 17, doi:10.3390/s17092022.

[165]Cheng, X.; Zhang, Y.; Chen, Y.; Wu, Y.; Yue, Y. Pest identification via deep residual learning in complex background. Computers and Electronics in Agriculture 2017, 141, 351–356, doi:10.1016/j.compag.2017.08.005.

[166]Espinoza, K.; Valera, D.L.; Torres, J.A.; López, A.; Molina-Aiz, F.D. Combination of image processing and artificial neural networks as a novel approach for the identification of Bemisia tabaci and Frankliniella occidentalis on sticky traps in greenhouse agriculture. Computers and Electronics in Agriculture 2016, 127, 495–505, doi:10.1016/j.compag.2016.07.008.

[167]Ding, W.; Taylor, G. Automatic moth detection from trap images for pest management. Computers and Electronics in Agriculture 2016, 123, 17–28, doi:10.1016/j.compag.2016.02.003.

[168]MIGUEL, S.; MOYA-ANEGÓN, F.; HERRERO-SOLANA, V. El análisis de co-citas como método de investigación en Bi-bliotecología y Ciencia de la Información. Investigación Bibliotecológica: archivonomía, bibliotecología e información 2007, 21, 139–155, doi:10.22201/iibi.0187358xp.2007.43.4129.

[169]Qin, X.; Han, X. Climate-Smart Agriculture in China: Current Status and Future Perspectives. Sustainability Sciences in Asia and Africa 2023, 205–231, doi:10.1007/978-981-99-2828-6_10.

[170]Sun, N.; Fan, B.; Ding, Y.; Liu, Y.; Bi, Y.; Seglah, P.A.; Gao, C. Analysis of the Development Status and Prospect of China's Agricultural Sensor Market under Smart Agriculture. Sensors 2023, 23, 1–15, doi:10.3390/s23063307.

[171]Khan, N.; Ray, R.L.; Sargani, G.R.; Ihtisham, M.; Khayyam, M.; Ismail, S. Current progress and future prospects of agri-culture technology: Gateway to sustainable agriculture. Sustainability (Switzerland) 2021, 13, 1–31, doi:10.3390/su13094883.

[172]Chandra, A.; McNamara, K.E.; Dargusch, P. Climate-smart agriculture: perspectives and framings. Climate Policy 2018, 18, 526–541, doi:10.1080/14693062.2017.1316968.

[173]T. S. Kondo and S. A. Diwani, "Artificial intelligence in Africa: a bibliometric analysis from 2013 to 2022," Dec. 01, 2023, *Springer Nature*. doi: 10.1007/s44163-023-00084-2.

[174]K. Gikunda, "Harnessing Artificial Intelligence for Sustainable Agricultural Development in Africa: Opportunities, Challenges, and Impact," Jan. 2024. Available: http://arxiv.org/abs/2401.06171

[175]D. C. Rodríguez-Lira, D. M. Córdova-Esparza, J. M. Álvarez-Alvarado, J. Terven, J. A. Romero-González, and J. Rodríguez-Reséndiz, "Trends in Machine and Deep Learning Techniques for Plant Disease Identification: A Systematic Review," Dec. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/agriculture14122188.

Evaluating User Experience in a Public Sector Digital System Through Nielsen's Heuristic Approach

Nagyan Yosse Wibisono[1], Viany Utami Tjhin[2]
Department Information Systems Management, Jakarta, Indonesia[1]
BINUS Graduate Program—Master of Information System Management, Bina Nusantara University, Indonesia[2]

***Abstract*—The digital transformation in public administration has encouraged the Indonesian National Police (Polri) to adopt a digital government application for managing official documents electronically. Despite its functional benefits, users have reported several usability issues such as non-intuitive navigation, inconsistent interface design, inadequate system feedback, and insufficient documentation. To systematically address these problems, this study combines Jakob Nielsen's Heuristic Evaluation (HE) with Partial Least Squares Structural Equation Modeling (PLS-SEM), offering a hybrid methodological approach that is rarely applied in public sector UX evaluation. Data were collected through a structured questionnaire distributed to 156 active users of the application. The instrument measured ten heuristic principles and user experience dimensions using a four-point Likert scale. The results reveal that all heuristic dimensions scored within the "good" range with mean values between 3.19 and 3.38, classified as cosmetic issues under Nielsen's severity scale. More importantly, the analysis shows that only three heuristics—Match Between System and the Real World, Help Users Recognize Diagnose and Recover from Errors, and Help and Documentation—have a significant positive impact on user experience perceptions. Together, these heuristics explain 90.9 per cent of the variance in user experience, highlighting their critical role in shaping user-centered digital government systems. This study advances existing evaluation models by demonstrating the effectiveness of integrating heuristic evaluation with quantitative SEM-based analysis, bridging diagnostic insights with statistical rigor. The findings provide a prioritized roadmap for improving the application's interface and emphasize the importance of user-centered design for enhancing the adoption and effectiveness of public sector digital systems.**

Keywords*—Usability evaluation; heuristic evaluation; user experience; digital government; PLS-SEM; public sector technology; UX mining; usability modeling

I. Introduction

Digital transformation has become a critical agenda in modern public sector organizations, particularly in efforts to enhance efficiency, transparency, and service quality. The Indonesian National Police (Polri), as a key state institution, has also embraced this shift through the implementation of a digital government application aimed at managing internal administrative processes such as task letters and official memos. This application replaces conventional paper-based workflows with an integrated digital platform, allowing for faster document processing, electronic validation, and improved traceability of administrative activities. Despite the functional advantages offered by such systems, their adoption and long-term success depend heavily on usability and the overall user experience (UX). Prior studies indicate that inadequate usability can significantly hinder system acceptance, reduce user satisfaction, and increase operational errors in public service platform [1][2]. In the context of this digital application used by Polri personnel, several recurring issues were identified through preliminary user feedback: non-intuitive navigation, inconsistency across interface layouts, insufficient system feedback, and the absence of clear user documentation. These problems, although often overlooked during development, directly affect user efficiency and satisfaction, especially in mission-critical environments such as law enforcement institutions.

To regularly evaluate the system's usability and understand how it impacts the user experience, this study employs the Heuristic Evaluation (HE) framework introduced by Jakob Nielsen. The ten usability heuristics provide a comprehensive lens through which interface design can be assessed, identifying both minor and severe issues related to user interaction. To complement this qualitative framework with empirical rigor, this study integrates HE with Partial Least Squares Structural Equation Modeling (PLS-SEM), a powerful multivariate technique suitable for analyzing complex models involving latent variables. This combination allows researchers to both diagnose usability problems and quantify their influence on the perceived UX.

Existing literature has demonstrated the effectiveness of HE and PLS-SEM separately, yet few studies have attempted to integrate both methods in evaluating digital government applications. Moreover, limited research has explored this approach within the unique operational context of Indonesian law enforcement institutions. Therefore, this study addresses a significant research gap by applying a hybrid HE–PLS-SEM approach to assess the usability of a digital application used by Polri personnel. The outcome is expected to offer a robust model that explains the impact of each usability dimension on UX, along with severity-based prioritization of interface improvements.

The study not only contributes methodologically by combining heuristic diagnostics with structural modeling but also provides practical insights for public sector developers and decision-makers. By identifying which usability principles most strongly affect user experience, the findings can inform future design and policy decisions aimed at enhancing the effectiveness of digital governance tools.

Research Questions

Based on the identified research problem, this study is guided by the following research questions:

- What usability issues are experienced by users of the Indonesian National Police's digital document system (Astina) as identified through Nielsen's heuristic evaluation?
- Which heuristic principles have the most significant influence on user experience, as measured using PLS-SEM?
- How can the findings be used to provide practical recommendations and a prioritized roadmap for improving the system's usability and supporting digital transformation in public sector organizations?

II. Liteaturew Review

A. User Experience

User Experience (UX) is an analytical process that focuses on understanding how users interact with digital products or services, including their perceptions, emotions, and levels of satisfaction during usage. UX exploration aims to identify user needs, challenges, and expectations towards a system, serving as the foundation for designing more intuitive, efficient, and user-centered products [3].

This process involves identifying desired features, analyzing issues or obstacles encountered during usage, and evaluating user satisfaction across various system aspects. Gathering user feedback is a key component in optimizing product design, enhancing the overall experience, and ensuring that the system is more responsive and efficient in meeting user needs [4].

Improving UX offers several benefits, including increased user satisfaction and loyalty, refinement of products or services to better align with user expectations, and enhanced effectiveness and efficiency of system usage. Furthermore, a deep understanding of user needs can foster innovation and generate new ideas that are more relevant to the market, resulting in continuous improvements and added value for users [5].

B. User Interface

User Interface (UI) plays a vital role in shaping the overall customer experience, acting as the visual and interactive layer between users and digital systems. In computing environments, the three key components—hardware, software, and human users (brainware)—are interconnected through the UI, which serves as the bridge that enables effective communication and interaction [6].

UI is central to the implementation of user-centered design (UCD), a fundamental principle of UI/UX that emphasizes placing users at the core of the design process. This approach is especially critical in financial applications, where usability and trust heavily influence the user experience. UCD involves deeply understanding user needs, behaviors, and constraints to design interfaces that address those elements effectively. In financial contexts, this might include simplifying complex financial tasks, presenting information clearly, and ensuring intuitive navigation [7].

Functionally, UI comprises two main components: input and output. Input elements involve user commands delivered through devices like keyboards or mice, while output components deliver system responses via visuals, sounds, or other feedback mechanisms. An effectively designed UI enhances user experience, improves system efficiency, and ensures ease of access to and comprehension of information [8].

C. Heuristic Evaluation

UX heuristics are a set of principles used to evaluate user interface designs to ensure they are intuitive, efficient, and user-friendly. These guidelines are commonly applied in heuristic evaluations to identify usability issues without direct user testing The goal is to improve interface elements based on user needs and expectations, facilitating efficient task completion and enhancing overall user satisfaction [8].

Key heuristics include [9]:

- Visibility of system status – keeping users informed about system activities through timely feedback.
- Match between system and the real world – using familiar language and conventions to ensure intuitive interactions.
- User control and freedom – providing clearly marked exits to prevent user frustration from unintended actions.
- Consistency and standards – aligning with platform conventions to reduce cognitive load (Jakob's law).
- Error prevention – designing to prevent problems before they occur through validation and confirmation.
- Recognition rather than recall – minimizing memory load by keeping key information visible.
- Flexibility and efficiency of use – allowing expert users to use shortcuts while supporting novice users.
- Aesthetic and minimalist design – presenting only relevant information to enhance clarity and usability.
- Help users recognize, diagnose, and recover from errors – providing user-friendly messages and solutions.
- Help and documentation – offering accessible, task-oriented help when needed.

These heuristics contribute to designing systems that are not only functional but also provide a positive and satisfying user experience [10].

III. Objectives

The primary objective of this study is to evaluate the usability and user experience (UX) of a digital government application implemented by the Indonesian National Police (Polri) to support internal administrative processes. Specifically, the research aims to:

1) Determine the severity rating level of each usability problem found in the application, as a basis for prioritizing system improvements.

2) Analyze the influence of usability aspects on the user experience (UX) of users in using the Astina application.

3) Identify and categorize significant usability problems based on the perceptions of active users of the Astina application.

Through these objectives, this study aims to offer a validated usability assessment model that can be applied to other digital government systems, while also supporting the adoption of user-centered design principles in public sector digital transformation.

IV. METHODS

A. Research Design

This study employed a quantitative research design using a survey-based approach to evaluate the usability and user experience (UX) of a digital government application used by the Indonesian National Police (Polri). The research integrated Heuristic Evaluation (HE) principles developed by Jakob Nielsen with Partial Least Squares Structural Equation Modeling (PLS-SEM) to statistically test the relationships between usability factors and user experience outcomes.

B. Instrument Development

The research instrument was a structured questionnaire consisting of 33 items grouped under 11 latent variables: 10 variables represented Nielsen's usability heuristics:

One variable captured user experience (UX), measured by three indicators: perceived ease of use, satisfaction, and efficiency. Each heuristic variable was operationalized using three indicators, making a total of 30 indicators for the usability construct. Responses were recorded using a 4-point Likert scale (1 = strongly disagree, 4 = strongly agree), selected to encourage decisive responses and avoid neutrality.

The instrument was validated by domain experts and tested in a pilot study involving 10 users. Subsequently, the internal consistency and construct validity were confirmed through Cronbach's Alpha and Pearson correlation analyses.

C. Sampling and Data Collection

The population consisted of active users of the digital application within Polri. Using purposive sampling, data were collected from 156 respondents who met the inclusion criteria:

Had used the application at least once for official administrative tasks; Were willing to participate and complete the survey fully. The survey was administered online using Google Forms, and data collection took place over two weeks in April 2025. This approach was practical for reaching respondents across various units within Polri and ensured secure and accessible participation.

D. Sampling and Data Collection

Severity Rating

Usability issues were categorized using Nielsen's severity rating scale (0–4), adapted for mean scores on the Likert scale:

0 = no usability problem (mean > 3.4)

1 = cosmetic problem

2 = minor usability problem

3 = major usability problem

4 = critical usability issue (mean < 1.59)

Each heuristic indicator's mean score was converted into a severity level to prioritize improvement areas.

Structural Equation Modeling (PLS-SEM)

PLS-SEM was conducted using SmartPLS v4 to assess both measurement and structural models:

- Outer model: Evaluated using convergent validity (outer loadings ≥ 0.7, AVE ≥ 0.5), discriminant validity (Fornell–Larcker), and reliability (Cronbach's Alpha ≥ 0.7).
- Inner model: Assessed using path coefficients, R^2 values, and bootstrapping (5,000 resamples) to determine the significance of relationships between heuristic variables and UX.

This two-stage analysis approach ensured that the model was both statistically sound and meaningful for theoretical and practical applications.

V. RESULT

A. Severity Rating

The average severity rating results for each heuristic dimension (H1–H10) are also in the same category, with a score value between 3.19 and 3.29. All dimensions received a severity rating of 1, which means that no minor, major, or critical problems were found in the main aspects of the application interface. A summary of the severity rating of all variables is presented in:

Several indicators that have scores close to the upper limit, such as HE12 (3.30) and HE23 (3.38) indicate very positive user perceptions of the elements, approaching a problem-free condition. Meanwhile, indicators with the lowest scores, such as HE102 (3.14) and HE103 (3.19) remain in the "good" category, but can be considered for quality improvement if you want to achieve an optimal user experience. The average severity rating results for each heuristic dimension (H1–H10) are also in the same category, with a score value between 3.19 and 3.29. All dimensions get a severity rating of 1, which means that no minor, major, or critical problems were found in the main aspects of the application interface. A summary of the severity ratings for all variables is presented in Tabel I.

B. PLS-SEM Analysis

The methodological procedure followed in this study included the distribution of structured questionnaires to 156 respondents, followed by data screening and coding. In the PLS-SEM analysis, the measurement (outer) model was first evaluated through convergent validity, discriminant validity, and reliability tests using indicators such as factor loadings, Average Variance Extracted (AVE), Composite Reliability (CR), and Cronbach's Alpha. Next, the structural (inner) model

was assessed through path coefficients, the coefficient of determination (R Square), predictive relevance (Q Square), and effect size (f Square). Finally, bootstrapping with 5,000 resamples was employed to test the significance of the hypothesized relationships (Table I).

TABLE I. SEVERITY RATING

Code	Average Score	Quality	Severity	Interpretation
H1	3,29	Good	1	Cosmetic Problem
H2	3,29	Good	1	Cosmetic Problem
H3	3,27	Good	1	Cosmetic Problem
H4	3,24	Good	1	Cosmetic Problem
H5	3,26	Good	1	Cosmetic Problem
H6	3,29	Good	1	Cosmetic Problem
H7	3,20	Good	1	Cosmetic Problem
H8	3,22	Good	1	Cosmetic Problem
H9	3,20	Good	1	Cosmetic Problem
H10	3,19	Good	1	Cosmetic Problem

Partial Least Squares Structural Equation Modeling (PLS-SEM) is used in this study to analyze the relationship between Nielsen's 10 usability heuristic principles (HE1–HE10) and the perception of User Experience (UX) in Astina application users. This method was chosen because PLS-SEM is very suitable for predictive and complex models and can accommodate a relatively moderate number of samples with a strictly non-normal data distribution. PLS-SEM analysis is carried out through two main stages, namely: evaluation of the measurement model (outer model) and evaluation of the structural model (inner model). In addition, hypothesis testing is also carried out to determine the effect of each usability heuristic variable on UX.

Evaluation of the outer model aims to ensure that the indicators used can reflect the constructs being measured in a valid and reliable manner. The outer model test covers four main aspects: indicator reliability, construct reliability, convergent validity, and discriminant validity.

The reliability of an indicator in a reflective measurement model refers to the ability of an indicator to reflect a latent construct consistently and accurately. In PLS-SEM analysis, indicator reliability is indicated by the outer loading value of each item against its construct in Table II.

According to [11], an indicator is said to have adequate individual reliability if it has an outer loading value ≥ 0.70. A value above 0.70 indicates that more than 50% of the variance in the indicator can be explained by the latent construct. If the loading value is between 0.40–0.70, the indicator can be considered to be retained if the overall construct remains valid and reliable. However, indicators with loading values below 0.40 are generally recommended to be deleted.

In this study, the outer loadings results show that all indicators in the model have loading values between 0.800 and 0.905 (Table II) which means:

- All indicators pass the minimum threshold of 0.70, so they can be said to be very reliable.
- There is no indicator that is weak or needs to be eliminated from the model.
- The existence of each indicator contributes significantly to forming the relevant latent construct.

TABLE II. INDICATOR RELIABILITY (OUTER LOADING)

Indicator	Construct	Outer Loading	Result
HE11	HE1	0,865	Reliable
HE12	HE1	0,868	Reliable
HE13	HE1	0,846	Reliable
HE21	HE2	0,890	Reliable
HE22	HE2	0,800	Reliable
HE23	HE2	0,875	Reliable
HE31	HE3	0,874	Reliable
HE32	HE3	0,860	Reliable
HE33	HE3	0,853	Reliable
HE41	HE4	0,873	Reliable
HE42	HE4	0,872	Reliable
HE43	HE4	0,865	Reliable
HE51	HE5	0,831	Reliable
HE52	HE5	0,832	Reliable
HE53	HE5	0,853	Reliable
HE61	HE6	0,867	Reliable
HE62	HE6	0,896	Reliable
HE63	HE6	0,874	Reliable
HE71	HE7	0,882	Reliable
HE72	HE7	0,838	Reliable
HE73	HE7	0,868	Reliable
HE81	HE8	0,878	Reliable
HE82	HE8	0,881	Reliable
HE83	HE8	0,859	Reliable
HE91	HE9	0,880	Reliable
HE92	HE9	0,862	Reliable
HE93	HE9	0,881	Reliable
HE101	HE10	0,905	Reliable
HE102	HE10	0,876	Reliable
HE103	HE10	0,884	Reliable
UX1	UX	0,864	Reliable
UX2	UX	0,876	Reliable
UX3	UX	0,868	Reliable

These results indicate that each indicator functions well in measuring its respective construct. The very high correlation strength between the indicators and the latent construct provides a strong foundation that this data has high measurement quality, which is an important prerequisite before proceeding to the evaluation of the structural model and hypothesis testing.

Construct reliability is a measure to assess the internal consistency of the indicators that form a latent construct. This consistency shows how much the indicators together describe the construct they represent. In this study, construct reliability was evaluated using two main measures, namely Cronbach's Alpha and Composite Reliability (CR).

Cronbach's Alpha measures homogeneity between indicators and is a classic reliability indicator often used in quantitative research. A high alpha value indicates that the indicators are highly correlated with each other. According to (Hair Jr, 2021), a good Cronbach's Alpha value for both exploratory and confirmatory research is at least 0.60, while a value above 0.70 indicates adequate reliability.

Composite Reliability is considered a more accurate measure of construct reliability than Cronbach's Alpha, because it does not assume that all indicators have the same contribution to the construct. A good CR should have a minimum value of 0.70, and a value above 0.90 indicates very high reliability.

Based on the results of data processing using SmartPLS, the Cronbach's Alpha values ranged from 0.790 to 0.867, and the Composite Reliability values ranged from 0.877 to 0.918 for all constructs, as shown in Table III.

TABLE III. CONSTRUCT VALIDITY AND RELIABILITY

Construct	Cronbach's Alpha	Composite Reliability	AVE	Decision
HE1	0,823	0,895	0,739	Valid & Reliable
HE2	0,818	0,891	0,733	Valid & Reliable
HE3	0,828	0,897	0,744	Valid & Reliable
HE4	0,835	0,901	0,752	Valid & Reliable
HE5	0,790	0,877	0,704	Valid & Reliable
HE6	0,853	0,911	0,773	Valid & Reliable
HE7	0,829	0,898	0,745	Valid & Reliable
HE8	0,844	0,906	0,762	Valid & Reliable
HE9	0,847	0,907	0,764	Valid & Reliable
HE10	0,867	0,918	0,790	Valid & Reliable
UX	0,838	0,903	0,756	Valid & Reliable

All constructs in this model have Cronbach's Alpha values above 0.79 and Composite Reliability above 0.87, which means:

- Each latent construct is consistently measured by its indicators.
- No constructs need to be eliminated or revised because all have met the minimum construct reliability limit recommended in PLS-SEM research.
- High CR values (> 0.90) such as in HE10 and UX indicate that the indicators in the construct are very cohesive in measuring their latent concepts.

Thus, it can be concluded that all usability heuristic and user experience constructs in this model have very good internal consistency, and the results are suitable for use in the next structural analysis stage (inner model and hypothesis testing).

Convergent validity in PLS-SEM analysis is evaluated using the Average Variance Extracted (AVE) measure. According to, a construct is said to meet convergent validity if the AVE value is ≥ 0.50. This value indicates that at least 50% of the indicator variance can be explained by the latent construct it measures, while the rest comes from errors or other variables outside the model.

Based on the calculation results using SmartPLS, all constructs in the model — namely the 10 usability heuristic principles (HE1–HE10) and the User Experience (UX) variable — show very good AVE values, ranging from 0.704 to 0.790, as shown in Table III.

All constructs in this model meet the requirements for convergent validity because they have AVE values above 0.70, far exceeding the minimum threshold of 0.50. This shows that:

- Most of the variance of the indicators can be explained by the construct, not by measurement error or other variables.
- The indicators used have high correlation with each other and are cohesive in measuring the intended latent construct.
- The validity of the measurement model (outer model) as a whole is strong and can be relied on to continue to the evaluation of the relationship between constructs (inner model).

Thus, it can be concluded that all constructs in this research model have very good convergent validity, which is an important prerequisite to ensure that the results of the structural analysis (path analysis and hypothesis testing) are based on a valid measurement model.

Discriminant Validity in PLS-SEM analysis, discriminant validity is evaluated using the Fornell-Larcker Criterion, which compares the AVE square root value of each construct with its correlation value to other constructs. According to Fornell and Larcker (1981), a construct is said to have good discriminant validity if: The AVE square root value of a construct is greater than the correlation value between the construct and other constructs in the model.

Based on the results of data processing through SmartPLS, all constructs (HE1 to HE10 and UX) show AVE square root values (on the table diagonal) that are higher than the horizontal and vertical correlation values between other constructs, as shown in Table IV.

These results show that for each construct:

- The √AVE value (on the diagonal) is higher than the correlation value with other constructs (off the diagonal).
- This means that each construct is more closely related to its own indicators than to other constructs.

- This meets the Fornell-Larcker criteria and proves that the model has good discriminant validity.

Thus, it can be concluded that all constructs in this model are unique and empirically different from each other, and can be used independently in structural modeling.

TABLE IV. FORNELL LACKER CRITERION

Construct	HE1	HE2	HE3	HE4	HE5	HE6	HE7	HE8	HE9	H10	UX
HE1	0,859										
HE2	0,782	0,856									
HE3	0,722	0,771	0,863								
HE4	0,752	0,833	0,746	0,867							
HE5	0,767	0,819	0,789	0,770	0,839						
HE6	0,732	0,801	0,755	0,816	0,810	0,876					
HE7	0,767	0,785	0,812	0,812	0,817	0,811	0,863				
HE8	0,702	0,796	0,756	0,775	0,795	0,844	0,836	0,873			
HE9	0,763	0,799	0,816	0,772	0,812	0,788	0,861	0,822	0,874		
HE10	0,750	0,792	0,755	0,809	0,830	0,805	0,824	0,823	0,840	0,889	
UX	0,778	0,883	0,801	0,827	0,816	0,808	0,844	0,811	0,898	0,887	0,869

Thus, all indicators and constructs in this model are declared valid and reliable, making them suitable for use in the evaluation of subsequent structural models.

Structural Model Evaluation (Inner Model**).** The evaluation of the inner model aims to test the causal relationship between exogenous constructs (HE1–HE10) and endogenous constructs (UX), as well as to measure how much the exogenous constructs contribute to explaining the UX variable.

Coefficient of Determination (R^2). The coefficient of determination (R^2) is the main indicator in the evaluation of the structural model (inner model) used to assess the predictive ability of exogenous constructs against endogenous constructs in the research model. In the context of this study, the endogenous construct in question is User Experience (UX), while the exogenous constructs are the ten usability heuristic principles (HE1 to HE10) based on the Nielsen framework (1994).

The R^2 value describes the proportion of variance from the endogenous variable (UX) that can be explained simultaneously by the exogenous variables in the model. The higher the R^2 value, the greater the ability of the exogenous construct to explain the endogenous construct.

According to Hair et al. (2021), the interpretation of the R^2 value in the context of the PLS-SEM model is as follows:

$R^2 \approx 0.75 \rightarrow$ substantial (strong)

$R^2 \approx 0.50 \rightarrow$ moderate (moderate)

$R^2 \approx 0.25 \rightarrow$ weak (weak)

The calculation results using SmartPLS show that the R Square (R^2) value for the User Experience (UX) variable is 0.909, with an adjusted R^2 of 0.902. These results are presented in the following Table V.

TABLE V. COEFFICIENT OF DETERMINATION R^2 AND ADJUSTED R^2

Endogenous Construct	R^2	Adjusted R^2	Interpretation
UX	0,909	0,902	Substantial

The R^2 value of 0.909 indicates that 90.9% of the variance in User Experience (UX) perception can be explained by the ten usability heuristic principles used in this model (HE1 to HE10). The remaining 9.1% is explained by other factors outside the model, such as user personal characteristics, infrastructure quality, or organizational context that are not included in the research construct.

This value is very high and falls into the substantial category, so it can be concluded that this model has very strong predictive ability. In other words, the usability heuristic principles formulated by Nielsen have been empirically proven to be able to explain UX perceptions in the use of the Astina application as a whole.

The reliability of R^2 is also strengthened by the adjusted R^2 of 0.902, which corrects the R^2 value based on the number of predictor variables in the model. This shows that the prediction results do not experience overfitting, and remain stable even considering the complexity of the model.

These findings are a strong indication that the usability heuristic model can be used as an effective theoretical framework in explaining and predicting the user experience of internal organizational applications. The high coefficient of determination also strengthens the argument that the user-centered approach in interface design is very relevant and has a significant impact on user perceptions of digital systems.

Path Coefficient and Hypothesis Testing. After the measurement model is declared valid and reliable, the next stage is to evaluate the relationship between latent constructs in the structural model (inner model) through path coefficient analysis and hypothesis testing. This analysis aims to test the significance of the influence of each exogenous construct, namely the ten usability heuristic principles (HE1–HE10), on the endogenous construct, namely user experience (UX).

Each hypothesis is tested by looking at the values:

Path Coefficient (β): shows the direction and magnitude of the influence.

- t-statistic: measures the statistical significance of the influence
- p-value: determines whether the influence is significant ($p < 0.05$)
- Decision: whether the hypothesis is accepted or rejected

Hypothesis testing was carried out using a bootstrapping procedure of 5,000 subsamples, as recommended by Hair et al. (2021), to obtain a sampling distribution and estimate the significance of the path. The results of the path coefficient analysis and hypothesis testing are shown in Table VI.

TABLE VI. PATH COEFFICIENT ANALYSIS AND HYPOTHESIS TEST RESULT

Hypothesis	Path	Path Coefficient (β)	t-statistic	p-value	Decision
H1	HE1 → UX	-0,009	0,112	0.911	Rejeted
H2	HE2 → UX	0,378	6,658	0,000	**Accepted**
H3	HE3→ UX	0,027	0,458	0,647	Rejeted
H4	HE4→ UX	0,029	0,536	0,592	Rejeted
H5	HE5→ UX	-0,091	1,560	0,199	Rejeted
H6	HE6→ UX	0,010	1,65	0,869	Rejeted
H7	HE7→ UX	0,058	0,853	0,394	Rejeted
H8	HE8→ UX	-0,082	1,402	0,161	Rejeted
H9	HE9→ UX	0,361	5,760	0,000	**Accepted**
H10	HE10→ UX	0,334	6,133	0,000	**Accepted**

Of the ten hypotheses proposed, only three were accepted, namely:

- H2 (HE2 → UX). The path coefficient of 0.378 with p = 0.000, indicates that the principle of "Match between system and the real world" has a positive and significant influence on UX. This means that a system that uses terms and logic that are in accordance with the user's way of thinking will improve the perception of user experience.
- H9 (HE9 → UX). The path coefficient of 0.361 and p = 0.000, indicates that a system that helps users recognize, diagnose, and recover errors effectively has a strong influence on user experience.
- H10 (HE10 → UX). With a coefficient of 0.334 and p = 0.000, these results indicate that the existence of easily accessible documentation and assistance also contributes significantly to improving UX.

Meanwhile, the other seven principles (HE1, HE3–HE8) show coefficient values that are not statistically significant ($p > 0.05$), so the hypothesis related to these principles is rejected. This indicates that in the context of using the Astina application, these principles do not have a strong direct influence on the perception of user experience.

Effect Size (f^2). In addition to evaluating the significance of the influence through path coefficients and hypothesis testing, the PLS-SEM structural model analysis also needs to consider the magnitude of the contribution of each exogenous construct to the endogenous construct, which is measured through the effect size. In this context, effect size is used to determine how much each usability heuristic principle (HE1–HE10) contributes to explaining the User Experience (UX) variable.

The interpretation of the f^2 value according to Cohen (1988) is as follows:

$f^2 \geq 0.35$ → large effect

$0.15 \leq f^2 < 0.35$ → medium effect

$0.02 \leq f^2 < 0.15$ → small effect

$f^2 < 0.02$ → not significant

The results of the calculation of the effect size (f^2) of each construct on UX are presented in Table VII.

TABLE VII. RESULTS OF THE CALCULATION OF THE EFFECT SIZE (F^2) OF EACH CONSTRUCT ON UX

Exogenous Construct	f^2	Effect Size Category
HE1 (Visibility of system status)	0,000	Not significant
HE2 (Match between system and the real world)	0,306	**Medium**
HE3 (User control and freedom)	0,002	Not significant
HE4 (Consistency and standards)	0,002	Not significant
HE5 (Error prevention)	0,018	Small
HE6 (Recognition rather than recall)	0,000	Not significant
HE7 (Flexibility and efficiency of use)	0,006	Not significant
HE8 (Aesthetic and minimalist design)	0,014	Small
HE9 (Help users recognize, diagnose, and recover from errors)	0,247	**Medium**
HE10 (Help and documentation)	0,232	**Medium**

From the table above, it can be seen that only three constructs have a medium effect size, namely:

- HE2 (Match between system and the real world) with $f^2 = 0.306$.
- HE9 (Help users recognize, diagnose, and recover from errors) with $f^2 = 0.247$.
- HE10 (Help and documentation) with $f^2 = 0.232$.

The three principles are not only statistically significant, but also have a fairly strong contribution in explaining UX, so they can be considered as key factors in the perception of user experience of the Astina application.

In contrast, constructs HE5 and HE8 show small effect sizes (f^2 <0.02–0.015), while HE1, HE3, HE4, HE6, and HE7 have very low or insignificant effect sizes, indicating that removing these constructs from the model will not have a significant impact on the R^2 value of UX.

The effect size evaluation strengthens the findings from the hypothesis test, that only a portion of Nielsen's usability heuristic principles provide a significant contribution to the perception of the UX of application users. This finding emphasizes that in the context of organizational or public sector applications such as Astina, the aspects of system communication, error support, and documentation are the interface design elements that have the most impact on users. Thus, the development of similar systems should focus on improving the quality of these aspects so that the user experience can be optimally improved.

Predictive Relevance (Q^2). In addition to evaluating the coefficient of determination (R^2) and effect size (f^2), the inner model analysis in the PLS-SEM approach also includes testing predictive relevance (Q^2). This test is used to determine whether the structural model has better predictive ability for endogenous constructs compared to models without exogenous constructs. The Q^2 test is carried out using the blindfolding technique, a procedure that systematically removes some data and then estimates the missing data based on the model. The Q^2 value obtained will indicate how well the model predicts the observed values of endogenous variables. Interpretation of the Q^2 value according to Hair et al. (2021) are as follows:

$Q^2 > 0.35 \rightarrow$ large predictive ability

$0.15 < Q^2 \leq 0.35 \rightarrow$ moderate predictive ability

$0 < Q^2 \leq 0.15 \rightarrow$ small predictive ability

$Q^2 \leq 0 \rightarrow$ no prediction (bad model)

Based on the blindfolding results in SmartPLS, the Q^2 value for the endogenous construct User Experience (UX) is 0.656, as shown in Table VIII.

TABLE VIII. PREDICTIVE RELEVANCE Q^2

Exogenous Construct	SSO	SSE	Q^2	Interpretation
UX	468.000	161.057	0,656	High predictive

The Q^2 value of 0.656 indicates that the model has a very strong predictive capacity for the User Experience (UX) construct. This means that the exogenous variables used (HE1 to HE10) are collectively able to predict UX with a high level of accuracy, far above the minimum threshold set in the PLS-SEM methodology ($Q^2 > 0.35$).

Thus, this research model is not only statistically significant (through p-value) and has a meaningful variable contribution (through f^2), but is also predictively relevant in the context of real users of the Astina application.

This finding strengthens the validity of the model as a whole, that the relationship between usability heuristics and UX built in this study is not only structurally valid, but also predictively applicable. The high Q^2 value confirms that the principles of usability heuristics chosen are indeed appropriate for use in developing or evaluating similar information systems, because the model is able to predict well how users will assess their experience when using the application. Thus, this model can be used as a reliable measuring tool for developing interface design and evaluating UX in information systems based on the organization's internal needs

Path Model Visualization. Fig. 1 below is a visual representation of the relationship between constructs in the structural model that has been analyzed using PLS-SEM. This model displays ten usability heuristic constructs (HE1 to HE10) as exogenous variables that lead to the user experience (UX) construct as an endogenous variable. In the model above, it can be seen that each exogenous construct (HE1 to HE10) has one arrow direction towards UX. The thickness of the arrow (if displayed) and the number above the arrow indicate the magnitude of the influence (path coefficient) of each variable on UX. Based on the path analysis and the results of the previous hypothesis test, it can be identified that three constructs, namely: HE2 (Match between system and the real world), HE9 (Help users recognize, diagnose, and recover from errors), HE10 (Help and documentation) have a significant and positive influence on UX, as indicated by the high path coefficient value and p-value <0.05. Meanwhile, the other seven constructs (HE1, HE3–HE8) showed no significant influence, with small path coefficient values and p-values > 0.05, which are depicted in the model as arrows with values close to zero. This model illustrates that not all usability heuristic principles have the same influence on the perception of user experience in the context of internal applications such as Astina. Only principles related to effective system communication, error support, and availability of documentation significantly affect UX. The visualization of this model strengthens the interpretation of the previous quantitative results, and provides an intuitive picture that a system design approach that focuses on communication, help, and guidance greatly determines user perceptions of the quality of their interactions with the system.

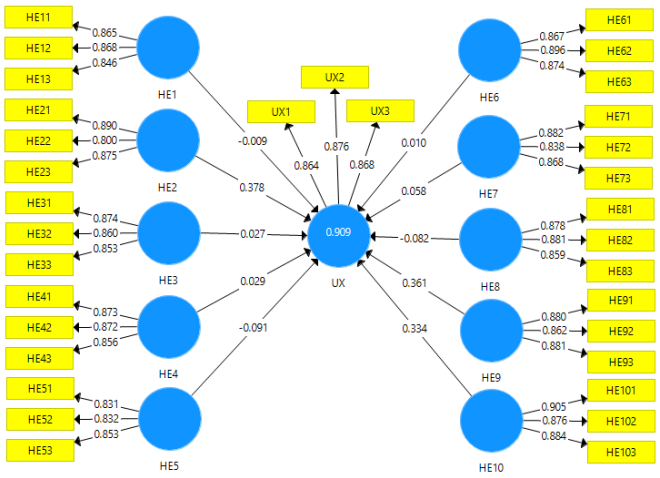


Fig. 1. PLS-SEM usability heuristic path model on UX.

VI. DISCUSSION

The results of this study aim to explain the influence of the usability heuristic principles developed by Nielsen (1994) on the perception of User Experience (UX) of Astina application users, which is a digital-based internal information system in the public sector. Testing was carried out using the PLS-SEM approach which allows simultaneous measurement between construct relationships and the predictive power of the model. Three

Heuristic Principles with Significant Influence on UX are discussed below.

A. Match Between System and the Real World

The HE2 construct has the highest path coefficient ($\beta = 0.378$, $p < 0.001$), indicating that the system's fit with the user's language, logical structure, and metaphor plays a major role in shaping UX perceptions. This principle emphasizes the importance of an interface that "speaks the user's language," as well as aligning interface elements with the user's real world to make it easier to understand (Nielsen, 1994).

This finding is supported who stated that user-language alignment is a key factor in reducing users' cognitive load and accelerating the adaptation of new systems. In the context of a public information system like Astina, where users come from a non-technical background, the use of familiar terms, intuitive icons, and organizational task-based navigation greatly helps improve the perception of experience. Research by [12] also showed that the difference between the technical terms of the system and the user's understanding can create interaction barriers, so that language alignment becomes essential in administrative systems.

Although HE2 has a statistically significant effect on UX, the severity rating results show an average score of 3.29, including the "Good" category, but it has not reached the "Very Good" category. This shows that users appreciate the conformity between the appearance and flow of the Astina system with the language and structure they understand; however, there are still minor shortcomings - such as the use of technical terms or symbols that are not yet fully intuitive - which cause them to rate the quality of the system as quite good but not yet ideal.

Implicitly, this means that users feel that this system is quite "familiar", but does not consistently reflect the user's everyday workflow or language. This reinforces the urgency of semantic and visual alignment between the system and the user's real world.

B. Help User Recognize, Diagnose, and Recover from Errors

The HE9 construct is also significant ($\beta = 0.361$, $p < 0.001$), indicating that the system's ability to handle user errors intelligently and informatively has a major impact on user experience. Users feel more confident and secure when the system is able to provide clear error information, as well as offer solutions or corrective steps.

System errors that cannot be handled directly by users are a major cause of frustration, and systems that fail to provide error feedback will hinder workflow. Meanwhile, Madzlan & Ishak (2024) added that good error recovery features (such as undo, warning dialogue, or error suggestion) are an integral component in building a resilient interface.

In a system like Astina, which is used in managing organizational administrative tasks, the existence of an error recovery system supports user efficiency and accuracy, and reduces the likelihood of fatal errors in the data input process.

HE9 also contributes significantly to UX, with an average severity rating score of the indicator between 3.16–3.22, which is included in the Good category. However, this score is one of the lowest among the other ten heuristic principles. This shows that although users feel the system helps them understand the errors that occur, there is still room for improvement, such as: clearer and more instructive error messages, automatic solutions or help that appears when an error occurs, undo or confirmation features before permanent action, Thus, HE9 is a critical element that is perceived as important, but has not been fully optimized in its implementation, as also emphasized by Azizi et al. (2021) in the health information system.

C. Help and Documentation

The HE10 construct also has a significant effect ($\beta = 0.334$, $p < 0.001$). Although this principle is usually considered a last resort, in the context of organizational applications with high complexity and moderate frequency of interaction, system documentation and assistance are still very much needed. Support in the form of tooltips, user guides, or interactive help systems provides a sense of security to new and old users.

Stated that accessible support systems not only help users solve problems but also foster the perception that the system is designed with empathy toward the user. This is also reinforced which shows that the availability of good documentation increases efficiency and perceived usability in hospital systems.

HE10 has a significant effect on UX and the average indicator score is 3.14–3.24, which is also included in the Good category, but is in the lowest range among other constructs. This means that even though help and documentation are considered important by users, current systems: may not provide enough contextual help, documentation is difficult to find or not available in the main interface, there is no searchable help or interactive guidance feature. This indicates a gap between expectations and implementation, where users consider that the existing help is limited or not easily accessible. This finding is in line with the study, which suggests that documentation must be on-demand, contextually relevant, and easy to navigate in order to truly contribute to UX.

Non-Significant Principles. The other seven principles (HE1, HE3–HE8) do not have a significant effect on the model. Although conceptually important, in the context of the Astina application, HE1 (Visibility of System Status) may be considered fulfilled, so that it is not a concern for users. Likewise, HE3 (User Control and Freedom), HE5 (Error Prevention), and HE8 (Aesthetic and Minimalist Design) may have less influence because the system is already quite standardized. In addition, HE6 (Recognition Rather Than Recall) and HE7 (Flexibility and Efficiency) may not have been optimized in terms of interface, so they have not created a direct effect on UX.

Research by [13] also found that not all usability heuristics have a direct influence on UX, depending on the application context and user characteristics. This means that theoretically important principles may have a statistically low influence if they are considered "default" by users.

Although only three of Nielsen's ten usability heuristic principles (HE2, HE9, and HE10) were proven to have a statistically significant effect on UX in the structural model, the other principles cannot be ignored. This is because the severity rating results show that all principles get an average score above

3.15, or fall into the "Good" category, with a severity classification at level 1 (cosmetic problems).

HE1 is not significant to UX (β = –0.009, p = 0.911), but gets a severity rating score of 3.29, indicating Good quality. This may be because the Astina system has provided quite good status indicators, such as notifications, loading status, or system messages, so that users are not too aware of their function explicitly because they are considered standard features. In other words, the absence of pain is not always the presence of pleasure — users do not feel disturbed, but also do not feel significantly helped.

HE3 is also not significant (β = 0.027, p = 0.647), but gets a rating score of 3.27. This shows that users value the freedom to navigate the system, for example through the back button, cancel, or menu selection. However, because the frequency of critical events that require this control may be low, users do not feel that the presence of this feature has a major impact on their experience.

HE4 obtained a small coefficient (β = 0.029, p = 0.592), with a severity score of 3.24, still in the "Good" category. This result indicates that the system has been quite consistent visually and functionally, but users may be less aware of minor inconsistencies, or insensitive to design standards, so that although theoretically important, it is not perceived as a dominant UX factor in the context of this application.

HE5 recorded a negative coefficient (β = –0.091, p = 0.119), with a severity score of 3.26. This is interesting, because even though users felt the system was good at preventing errors, they still valued the system more for providing solutions when errors occurred (HE9). In other words, error handling was perceived more than error prevention, especially in systems that were not very complex to use.

HE6 was also insignificant (β = 0.010, p = 0.869), with a severity score of 3.29. This suggests that users value interfaces that help them access information without having to remember a lot of things, such as dropdowns, labels, and icons. However, because these features run "behind the scenes," their contribution to UX is not considered dominant in their overall assessment.

HE7 had a low path coefficient (β = 0.058, p = 0.394), and a severity score of 3.20. In a system like Astina that is administrative and standardized, flexibility and efficiency may not be the main needs of users, so even though the quality is good, its impact on the overall UX is not significant. HE8 recorded a negative coefficient (β = -0.082, p = 0.161), but the severity score of 3.22, remains at the "Good" level. A clean and uncomplicated system does increase user comfort, but in a system with dominant functional functions, aesthetic appearance has not been a major factor in user satisfaction or comfort [14]. This explains why the perception remains positive, but its impact on UX is not statistically significant.

VII. Conclusion and Contribution

This study contributes methodologically by combining Nielsen's Heuristic Evaluation, which is widely used for diagnosing usability issues qualitatively, with Partial Least Squares Structural Equation Modeling, which provides quantitative validation of causal relationships. Compared to traditional heuristic evaluation that mainly identifies problems without measuring their statistical significance, the hybrid approach allows for both problem identification and empirical testing of their influence on user experience. In contrast to other UX evaluation models such as the System Usability Scale (SUS), User Experience Questionnaire (UEQ), or the Technology Acceptance Model (TAM), which focus on overall user perceptions, the HE–PLS-SEM integration offers a more detailed diagnostic framework. It not only highlights which usability heuristics matter most but also quantifies their contribution to UX outcomes, thereby advancing existing evaluation methods in both rigor and practical applicability for public sector systems.

The originality of this study lies in the integration of heuristic evaluation with PLS-SEM in evaluating a government application. While both methods have been widely applied independently in prior UX studies, their combination in the context of public sector systems, particularly the Indonesian National Police application, has not been previously explored. Prior research using heuristic evaluation has mainly produced qualitative insights into usability issues without quantifying their statistical significance, whereas studies applying PLS-SEM have typically examined user perceptions using generic acceptance models such as TAM or UTAUT without direct diagnostic links to usability heuristics. By merging these two approaches, this study not only identifies usability problems but also validates their causal impact on user experience outcomes. This represents a novel contribution, offering both methodological advancement and practical guidance for improving digital government systems.

Data Availability

Dataset is available from the Zenodo Repository, DOI: 10.5281/zenodo.16890311.

CRediT authorship contribution statement

Nagyan Yosse Wibisono: Conceptualization, Methodology, Formal Analysis, Investigation, Writing – Original Draft. Viany Utami Tjhin Supervision, Project administration, Funding acquisition, Writing – Review & Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Azizi, A., Maniati, M., Ghanbari-Adivi, H., Aghajari, Z., Hashemi, S., Hajipoor, B., Qolami, A. R., Qolami, M., & Azizi, A. (2021). Usability evaluation of hospital information system according to heuristic evaluation. *Frontiers in Health Informatics*, *10*. doi: 10.30699/fhi.v10i1.271

[2] Galavi, Z., & Khajouei, R. (2023). Proposing an initial model for the heuristic evaluation of mHealth applications. *Informatics in Medicine Unlocked*, *41*. doi: 10.1016/j.imu.2023.101322

[3] Pei, X., Fu, S., & Jiang, T. (2023). An empirical study on user experience evaluation of VR interface in digital museums. *Data and Information Management*, *7*(4). doi: 10.1016/j.dim.2023.100057

[4] Lehnert, F., Doublet, S., & Sim, G. (2025). Designing usability/user experience heuristics to evaluate e-assessments administered to children. *Computer Standards & Interfaces*, *92*, 103933. doi: 10.1016/J.CSI.2024.103933

[5] Macias-Velasquez, S., Medellin-Castillo, H. I., & Garcia-Barrientos, A. (2024). New-user experience evaluation in a semi-immersive and haptic-enabled virtual reality system for assembly operations. *International Journal of Human-Computer Studies*, *190*, 103312. doi: 10.1016/J.IJHCS.2024.103312

[6] Faudzi, M. A., Cob, Z. C., Ghazali, M., Omar, R., & Sharudin, S. A. (2024). User interface design in mobile learning applications: Developing and evaluating a questionnaire for measuring learners' extraneous cognitive load. *Heliyon*, *10*(18), e37494. doi: 10.1016/J.HELIYON.2024.E37494

[7] Quiñones, D., Ojeda, C., Herrera, R. F., & Rojas, L. F. (2024). UXH-GEDAPP: A set of user experience heuristics for evaluating generative design applications. *Information and Software Technology*, *168*, 107408. doi: 10.1016/J.INFSOF.2024.107408

[8] International Standard. (2019). Ergonomics of human-system interaction-Human-centred design for interactive systems COPYRIGHT PROTECTED DOCUMENT. Retrieved from www.iso.org

[9] Nielsen Norman Group (2025) 10 Usability Heuristics for User Interface Design

[10] Pierre, R. da S. de Q. (2015). Heuristics in Design: A Literature Review. *Procedia Manufacturing*, *3*, 6571–6578. doi: 10.1016/J.PROMFG.2015.07.961

[11] K. Eli Hair Jr. (2021). Classroom Companion: Business Partial Least Squares Structural Equation Modeling (PLS-SEM) Using R AAWorkbook.ssa, "Title of paper if known," unpublished.

[12] Bouraghi, H., Rezayi, S., Amirazodi, S., Nabovati, E., & Saeedi, S. (2022). Evaluating the usability of a national health information system with heuristic method. *Journal of Education and Health Promotion*, *11*(1), 182. doi: 10.4103/jehp.jehp_349_21

[13] Andika, R., & Renaldi, D. (2024). Heuristic Evaluation of UI/UX to Enhance Experience and Sales in E-commerce. *Bit-Tech*, *7*(2), 224–251. doi: 10.32877/bt.v7i2.1730

[14] Fatkhuri, Dharmawan, D., Desty Febrian, W., Karyadi, S., & Sani, I. (2024). Application of Heuristic Evaluation Method to Evaluate User Experience and User Interface of Personnel Management Information Systems to Improve Employee Performance. *Jurnal Informasi Dan Teknologi*, 14–20. doi: 10.60083/jidt.v6i1.466

NEBULA Framework: An Adaptive Framework for Unstructured Description to Solve Cold Start Problem

I Gusti Agung Gede Arya Kadyanan[1*], Ni Made Ary Esta Dewi Wirastuti[2],
Gede Sukadarmika[3], Ngurah Agus Sanjaya ER[4]
Doctoral Program Engineering Science-Faculty of Technic, Udayana University, Badung, Indonesia[1]
Department of Electro-Faculty of Technic, Udayana University, Badung, Indonesia[2, 3]
Department of Informatics-Faculty of Mathematics and Natural Science, Udayana University, Badung, Indonesia[4]

***Abstract*—The cold start problem is one of the main challenges in recommendation systems, especially when the system has to provide recommendations for new items that do not yet have a history of interaction. Although various approaches have been developed, most still use conventional interaction-based methods, which are not optimal in providing accurate recommendations for new items that only have minimal and unstructured descriptive information. This research aims to provide recommendations for new items that lack interaction history and have unstructured descriptive information by addressing the cold start problem more adaptively. The proposed model is based on Named Entity Recognition (NER) and metadata representation as an adaptive framework capable of adjusting recommendation methods based on the availability of initial information. For new items, the system utilizes basic attributes such as product type, materials, and origin, and employs an adaptive approach for rating prediction. Testing results demonstrate system performance with an Accuracy of 0.967, Precision of 0.838, Recall of 0.846, F1-score of 0.842, and an average Mean Absolute Error (MAE) of 0.159. This adaptive framework proved to be superior to conventional approaches, with improvements in Precision of 15.59%, Recall of 17.50%, F1-score of 16.54%, and a significant reduction in MAE. Additionally, the Kappa value of 0.69 indicates a high level of agreement (substantial agreement) among validators. These findings demonstrate that the system is not only more accurate in recommending new items but also more reliable under minimal data conditions, thereby enhancing user confidence. Overall, this NER and metadata-based framework can serve as an effective solution for addressing the cold start problem and improving recommendation quality during the initial stages.**

Keywords*—Cold start; adaptive framework; recommender system; NER; unstructured description

I. Introduction

Recommendation systems have become an important component in various modern digital platforms, including e-commerce, media services, tourism, and culture. These systems help users filter information and find relevant products or services based on personal preferences, thereby improving the efficiency of interactions and overall user satisfaction [1], [2], [3].

However, one of the main challenges in recommendation systems is the cold start problem, which occurs when new items or users lack sufficient historical data, such as ratings or interaction history. This situation makes it difficult for traditional recommendation algorithms, which generally rely on previous interaction patterns [4]. The cold start problem can arise in various scenarios, such as when new products are added to the system, new users join, or items have limited descriptive data [5], [6], [7].

The cold start problem is particularly critical because it directly impacts the exposure and accessibility of new products. In commercial sectors such as consumer goods, fashion, technology, or daily necessities, new items often fail to receive optimal recommendations despite having high market potential. Existing recommendation systems still heavily rely on historical interaction data, causing new products to be less visible to users [8]. This situation highlights a research gap: the need for an approach that can leverage descriptive information or metadata more adaptively, rather than solely relying on interaction history.

Previous studies have attempted to address the cold start problem using different approaches. For example, a hybrid collaborative filtering method combined with content-based filtering [9], or the use of NLP-based descriptive representations [10], [11]. However, most of these studies still face limitations in terms of metadata quality and difficulties in extracting information from minimal and unstructured product descriptions. Additionally, existing models are generally rigid and not fully adaptive to data variations in specific domains, resulting in system performance degradation when faced with new products with incomplete or inconsistent metadata. This highlights a research gap, namely the need for a new approach that not only processes metadata but also possesses adaptive, flexible, and contextual capabilities in handling variations in product descriptions.

As a solution, this study proposes a metadata-based recommendation system model utilizing Natural Language Processing (NLP) with a focus on Named Entity Recognition (NER) techniques. This approach enables the system to extract important attributes from product descriptions, such as category, materials, and region of origin, to form vector representations that can be used in similarity calculations between items [12], [13]. To address conditions where no items exceed the similarity threshold, the system is also equipped with a clustering-based fallback strategy, ensuring that recommendations remain relevant. The model's strengths lie not only in its ability to address content-based cold start issues without relying on user interaction history, but also in its

*Corresponding author.

adaptive nature, which can adjust to limitations in descriptions, differences in writing style, and language variations in the domain of traditional crafts.

Thus, the objective of this research is to develop an adaptive framework based on metadata and NER to address the cold start problem for new items, especially when descriptive information is still limited. From a practical standpoint, the system is expected to increase the exposure of new products, such as traditional crafts, thereby contributing to local economic empowerment. From an academic standpoint, this research enriches the literature on NLP and metadata-based recommendation systems and offers a new contribution in the form of a more flexible and contextual adaptive approach, which opens up opportunities for wider application in various domains with similar challenges.

The contributions of this research can be summarized into three main aspects. First, from an algorithmic perspective, the proposed framework integrates multilingual Named Entity Recognition (NER) with adaptive embedding selection (spaCy for English and BERT for Indonesian), dynamic database switching, and clustering-based fallback mechanisms to ensure recommendation continuity under sparse conditions. Second, from a methodological perspective, this study advances the handling of unstructured product metadata by transforming free-form descriptions of traditional crafts into structured semantic entities that can be embedded and compared systematically. Third, from an application perspective, this research demonstrates the applicability of the framework to the domain of traditional crafts, showing how cultural product metadata can be leveraged to overcome cold start limitations while preserving local knowledge in digital marketplaces. Collectively, these contributions distinguish this research from existing NLP and content-based solutions, positioning the framework not merely as an applied refinement but also as a substantial advancement in adaptive cold start handling.

The remainder of this paper is organized as follows. Section II reviews related works that address the cold start problem in recommendation systems. Section III introduces the proposed metadata-based adaptive model with Named Entity Recognition. Section IV presents the model analysis and its components. Section V provides the performance assessment through experimental evaluation. Finally, Section VI concludes the paper and outlines potential directions for future research.

II. Related Work

The cold start problem has become one of the main challenges in the development of modern recommendation systems. Xiao et al. [14] through the UPRec approach, argue that a pre-training process that explicitly integrates user attributes can enrich user representations and reduce dependence on limited historical data, thereby effectively addressing cold start in sequential recommendation scenarios.

A similar approach was also comprehensively reviewed by Zhou et al. [15], who grouped deep learning-based methods such as BERT4Rec and Graph Neural Network (GNN) as cutting-edge solutions to sparsity and cold start problems, through the formation of deeper and more contextual user and item embeddings.

In the content and emotion-based domain, Jiang et al. [16] developed a recommendation system by combining Word2Vec and LSTM for sentiment analysis, demonstrating that even short reviews can be utilized to form user emotion modeling in social recommendation systems. An NLP-based approach was also adopted by Mishra et al. [11], who combined review classification using Random Forest and a collaborative filtering-based filtering model. The study demonstrated that review quality and text analysis can improve recommendation accuracy even under cold start conditions.

On the other hand, content-based filtering approaches relying on product metadata have been extensively developed for domains facing user interaction limitations. Negara et al. [17] built a recommendation system for an NFT marketplace using only item name and description features based on TF-IDF and cosine similarity, which proved capable of delivering relevant results even without any user ratings at all.

A metadata-based approach was also proposed by Lestari et al. [18] in the context of MSME marketplaces. They combined clustering and imputation techniques to fill data gaps in new products and showed that this combination was able to significantly improve prediction accuracy by up to 100% in KNN and Naive Bayes algorithms.

Widayanti et al. [9] also proposed a hybrid filtering-based recommendation system that combines Collaborative Filtering and Content-Based Filtering to overcome cold start and expand the diversity of recommendations. This hybrid strategy leverages the advantages of CF in detecting user preference patterns and the strength of CBF in explicitly extracting content features, thereby providing richer and more contextual predictions.

Panteli et al. [19] offer an approach based on frequent pattern mining and user clustering that can estimate new users preferences through distinctive purchasing patterns within each cluster of existing users. This model has proven successful in addressing the cold start problem, achieving a precision accuracy of 90.3% and doubling data density in experiments with MovieLens.

Gupta et al. [20] developed an adaptive e-commerce recommendation system using a hybrid deep learning-based approach, integrating Multilayer Perceptron (MLP), Matrix Factorization, autoencoder, and transformer-based re-ranking. This approach is highly effective in addressing cold start and data sparsity by leveraging various user behavior sources such as clicks, searches, and explicit ratings, achieving an accuracy performance of up to 91%.

Ramakrishna et al. [21] introduced the Hybrid Collaborative Filtering (HCoF) approach, which combines three filtering approaches: Collaborative, Social, and Semantic. This system successfully addresses the cold start problem on social networks by relying on similarity based on interests, location, and social trust levels. Evaluations show that this approach achieves an F1-score of 0.651 and high fairness across various user segments.

Afoudi et al. [22] combined Collaborative Filtering, Content-Based Filtering, and Self-Organizing Map (SOM) into a single hybrid system architecture. This model strengthens

user preference predictions based on genre and demographics and is capable of generating relevant recommendations even under data constraints or cold start conditions.

In addition to technical approaches, Vomberg et al. [23] discuss the cold start problem in the context of AI system implementation strategies that are still in their early stages. In their article, they emphasize the importance of building a data network effect to accelerate the data accumulation process in new systems. This approach focuses on organizational and business strategy aspects, indicating that solutions to the cold start problem are not always algorithmic but can also be achieved through system design that promotes gradual and targeted data growth.

From the various studies reviewed, it appears that the approaches to overcoming cold start problems in recommendation systems are very diverse, ranging from attribute-based pre-training, the use of deep learning embedding, NLP-based sentiment analysis, to hybrid methods that combine Collaborative Filtering (CF) and Content-Based Filtering (CBF). However, each study still has certain limitations, whether related to dependence on historical data, metadata quality, or adaptability to data variations in specific domains. To clarify the position of this research compared to previous works, Table I presents a summary of the comparison of approaches, advantages, disadvantages, and research gaps that this research attempts to fill.

Table I shows that previous studies have made important contributions to overcoming the cold start problem, but most are still limited to dependence on historical data, review quality, or limitations in adapting to product description variations. This study is different because it offers a more adaptive approach through the use of product metadata and Named Entity Recognition (NER) techniques, which enable the system to extract important attributes contextually. Thus, the main contribution of this study is to provide a solution for cold start items in traditional handicraft products, while enriching the literature on metadata-based recommendation systems and NLP in the local cultural domain.

TABLE I. COMPARISON OF PREVIOUS RESEARCH WITH THIS RESEARCH

Researcher & Year	Approach/Method	Strengths	Research Gap	Contribution of This Research
Xiao et al. (2023) [14]	UPRec with pre-training of user attributes	Enriches user representation, reduces reliance on historical data	Focuses on user-based cold start, does not address item cold start	Focus on item-based cold start using product metadata
Zhou et al. (2023) [15]	Deep learning (BERT4Rec, GNN)	More contextual user/item embedding, effective for sparsity	Requires large-scale data, less adaptive in narrow domains	Adaptive approach based on descriptive metadata
Jiang et al. (2020) [16]	Word2Vec + LSTM for sentiment analysis	Utilizes short reviews, captures user emotions	Relies on review quality, not always available	Independent from reviews, focused on product metadata
Mishra et al. (2024) [11]	NLP + Random Forest + Collaborative Filtering	Combines text and interaction, improves accuracy	Still requires interaction data, vulnerable to item cold start	Handles cold start without user interaction data
Negara et al. (2023) [17]	TF-IDF + Cosine Similarity (NFT marketplace)	Relevant even without ratings	Simple metadata, noisy, lacks deeper representation	Richer feature extraction with NER
Lestari et al. (2024) [18]	Metadata + clustering + imputation (MSME marketplace)	Fills missing data, significant accuracy improvement	Non-contextual, limited to specific domains	Adds local and cultural aspects
Widayanti et al. (2023) [9]	Hybrid CF + CBF	Provides higher recommendation diversity	Still relies on user interaction data	Independent from interaction history, adaptive metadata
Panteli et al. (2023) [19]	Frequent pattern mining + user clustering	Estimates new user preferences, precision 90.3%	Focuses on user cold start, not new items	Addresses cold start specifically for new items using descriptions
Gupta et al. (2024) [20]	Deep learning hybrid (MLP, MF, Autoencoder, Transformer)	Highly accurate (91%), leverages multi-source data	Complex, requires large-scale data (clicks, searches, ratings)	Lightweight, focused on descriptive metadata
Ramakrishna et al. (2023) [21]	Hybrid Collaborative Filtering (HCoF): CF + Social + Semantic	Effective in social networks, F1-score 0.651	Context-limited (social networks), still requires interest/location data	Local culture and product context without social interactions
Afoudi et al. (2021) [22]	CF + CBF + Self-Organizing Map (SOM)	Strengthens preference prediction, relevant in limited data scenarios	Complex, still dependent on genre/demographics	Focus on metadata of traditional handicraft products
Vomberg et al. (2023) [23]	AI implementation strategy (data network effect)	Accelerates accumulation of new data through system design	Does not offer algorithmic technical solution	Technical adaptive solution based on NLP (NER + metadata)
This Research	Metadata + NER + embedding + fallback clustering	Adaptive, cultural product focus, no historical data required	–	Tackles item cold start for handicraft products in context

III. PROPOSED MODEL

The proposed recommendation system model consists of three main stages, such as descriptive extraction, similarity estimation, and rating prediction. These three stages are summarized in the block diagram in Fig. 1, where each block shows the processing flow from new item input to rating estimation. This process is designed adaptively to address the cold start problem for new items that have no interaction history or user ratings [24]. Suppose a new item is denoted as

N, and the set of existing items in the database is denoted as $C = \{C_1, C_2, \ldots, C_j\}$, where each C_j represents a product that already has descriptive information and ratings from users. The framework used in this study is named NEBULA (Named Entity and BERT-based Framework for Unstructured Language Analytics). This name reflects three main components: Named Entity refers to the Named Entity Recognition (NER) module responsible for extracting semantic entities from unstructured product descriptions; BERT-based indicates the use of BERT embeddings to generate deeper contextual representations; and Unstructured Language Analytics emphasizes the framework's focus on processing free-form, non-standard product descriptions, which are the primary source of cold start problems. The choice of the name NEBULA also carries philosophical meaning, as in astronomy, a nebula is known as the "birthplace of stars". This metaphor aligns with the framework's purpose of transforming raw, unstructured data into meaningful knowledge, generating new insights, and producing relevant recommendations even under cold start conditions.

The dataset used in this study consists of 1,000 descriptions of traditional handicraft products collected from various marketplaces. Each product entry includes metadata such as product name, product description, and user ratings. As an illustration, Table II presents five sample product data obtained from the marketplace, which form part of the dataset used in this study.

Table II presents five sample product data obtained from the marketplace, consisting of product name, description, and rating. Product descriptions on marketplaces are generally written freely by sellers without following a standardized format. As a result, the text is considered unstructured, since it varies in writing style, vocabulary, and terminology, often leading to inconsistency. This poses a challenge in the text processing stage, making it necessary to apply Natural Language Processing (NLP) techniques to extract key entities such as product name, material, and origin. Moreover, unstructured descriptions may appear in both Indonesian and English, which requires the system to handle multilingual processing efficiently.

The annotation stage is carried out to label the main entities in the product description. The annotation process uses a Named Entity Recognition (NER) scheme with three types of entities, namely: Product (name or type of craft), Material (main material used), and Origin (region or location of origin of the product). Annotation is done manually by a team of annotators using the BIO (Begin, Inside, Outside) format to ensure consistency in entity tagging. An example of the annotation results can be seen in Fig. 2.

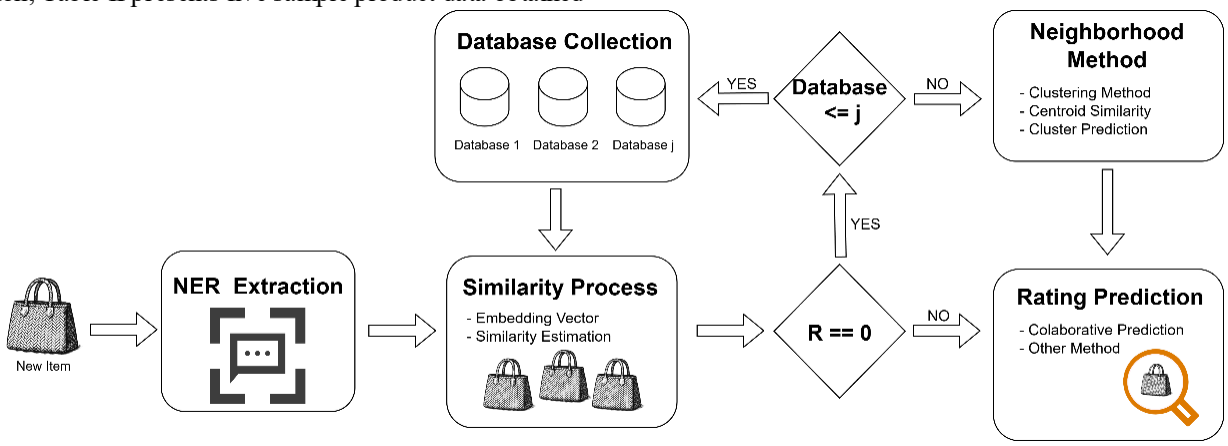


Fig. 1. Framework of the proposed recommendation rating.

TABLE II. SAMPLE DATASET

Product Name	Description	Rating
Ayam Antik Bambu	Ayam Antik, 30 cm, Bambu, merupakan patung dekoratif yang dipahat dari bambu kering dengan teknik ukir alami. Dibuat oleh seniman di Desa Batuan, cocok sebagai pemanis interior yang unik dan bernuansa etnik.	4.2
Gantungan Kunci Barong	Gantungan Kunci Barong ini dibuat dari kayu pule yang dipahat dan dilukis secara manual dengan detail wajah Barong. Produk ini berfungsi sebagai hiasan gantungan kunci yang mencerminkan budaya spiritual Bali. Warna dominan merah, hitam, dan emas memberikan kesan eksotis. Dibuat oleh pengrajin di Desa Mas, Ubud, yang terkenal akan keahlian ukiran kayunya.	4.6
Tali Tasel	Tali Tasel dari Tabanan, Bali, terbuat dari mendong asli yang berkualitas.	4.6
Duck-Shaped Wooden Pull Toy	This Wooden Pull Toy in the shape of a Duck is made from lightweight albasia wood and painted with bright, non-toxic colors. Produced by artisans from Desa Tegallalang, this toy is safe for children and evokes the nostalgia of traditional games.	4.7
Woven Rattan Bracelet	This woven rattan bracelet is made from authentic Balinese rattan, lightweight and comfortable to wear, suitable for enhancing your everyday look.	4.3

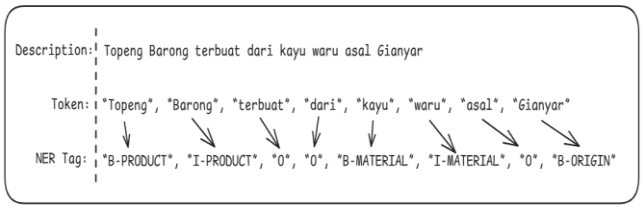


Fig. 2. Data annotation.

Next, the dataset is then split into training and testing sets using an 80:20 ratio, with performance evaluation conducted using precision, recall, and F1-score to measure the accuracy of entity extraction, and Mean Absolute Error (MAE) to evaluate rating prediction errors, while expert validation by two experts in recommendation systems and natural language processing was carried out to assess the relevance of recommendation results for new products.

In the first stage, when a new item N is added to the system, descriptive information about the product is processed using the Named Entity Recognition (NER) technique [25]. This descriptive information is text that contains explanations about the product and is the main source for extraction to determine the entities of the product. Through the NER process, the system identifies and classifies important entities in the text, such as product names, materials, and locations. The identified entities are then compiled into a set $E_N = \{e_1, e_2, \ldots, e_k\}$ that represents the characteristics of item N.

Next, the system accesses set C, which has also undergone the previous NER process, generating entity representation E_{C_j} for each item $C_j \in C$. The next stage is the similarity estimation process, in which the system converts the entity set into vector representations using the embedding method. The embedding method is selected adaptively based on data characteristics. For example, BERT (Bidirectional Encoder Representations from Transformers) is used for descriptions with complex contexts, especially in Indonesian, as it can understand complex sentence structures [26]. Meanwhile, spaCy is more suitable for general, lightweight entities, such as descriptions in English, as it is more efficient and faster [27]. Other alternative embedding methods can also be used depending on the needs and type of data being processed.

After that, the similarity score is calculated using the cosine similarity method, which was chosen for its simplicity and ability to effectively measure the closeness between text representations. An item C_j will be included in the reference set $R \subseteq C$ if it has a similarity value with the new item N that exceeds the set threshold, which is 0.70 [28]. Thus, R contains a collection of products in the database that are considered relevant as a reference for predicting the rating of new items.

After the reference set R has been determined, a check will be performed to see whether the set R is empty or not. If the set R has a value, the system will estimate the initial rating $\hat{r}_N$ for the new item N using the weighted average method of the ratings of the items in R, where the weight is given based on the similarity score of each item. However, if the reference set R is an empty set ($R = \emptyset$), the system will trigger the next adaptive mechanism, which is to switch to another database available in the system's database collection. The reanalysis process for items in the new database is performed starting from NER extraction, embedding, to similarity estimation to try to form a new reference set R. If items with sufficient similarity are found in this process, the rating prediction process can be performed immediately.

However, if all databases have been checked and no items with similarity values exceeding the threshold are found ($R = \emptyset$ overall), the system will activate a clustering-based fallback strategy. In this strategy, all existing products in the repository are clustered into several groups using methods such as Probabilistic Neural Network (PNN) or Generalized Lloyd Algorithm (GLA). Each cluster has a centroid vector calculated from the average embedding of the items in that cluster [29]. The new product N is then compared with all cluster centroids using cosine similarity, and the system selects the cluster with the highest similarity. After that, the system calculates the average rating of all products in the selected cluster [30]. This value is then used as the rating prediction for the new product N. The flowchart of the proposed method can be seen in Fig. 3 below.

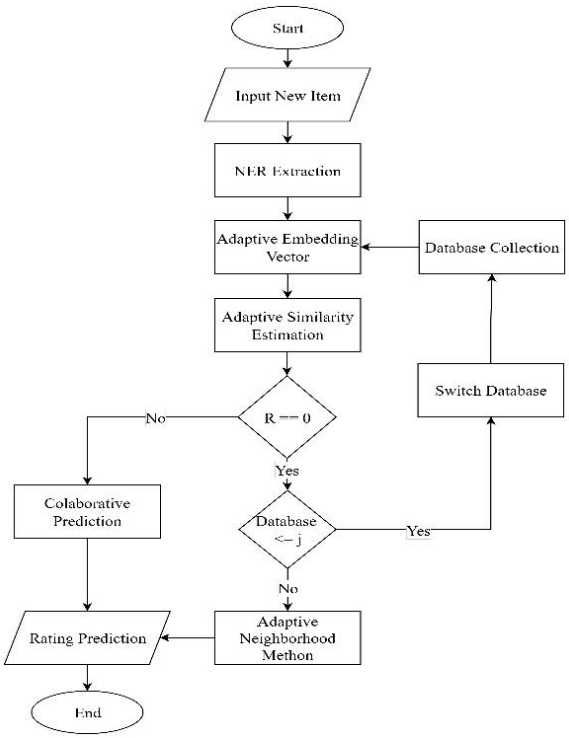


Fig. 3. The proposed model flowchart.

This adaptive mechanism constitutes the novelty of the NEBULA framework. Unlike prior content-based or metadata-driven cold start solutions, NEBULA explicitly integrates multilingual NER (spaCy for English and BERT for Indonesian), dynamic database switching when no similar items surpass the threshold, and a clustering-based fallback strategy to guarantee recommendation continuity. These three layers of adaptability ensure that the system remains robust under varying data conditions, addressing the cold start problem not only algorithmically but also strategically at the data and application levels.

IV. Model Analysis

A. Item Extraction

The first step in the proposed recommendation system is the Item Extraction process, which involves extracting descriptions to search for entities based on the metadata of traditional handicraft products. This process utilises Named Entity Recognition (NER) techniques to extract important entities such as product type (bags, statues, etc.), material type (wood, bamboo, fabric), and product origin location (Gianyar, Tabanan, Karangasem).

Each new product N that is entered will undergo this process to generate entities from the product, in the form of attribute-value pairs derived from the description text (E_N). This representation forms the basis for calculating similarity between products (Algorithm 1).

Algorithm 1: NER Extraction

```
Input: NewProduct
Output: ExtractedEntities
Function ExtractEntities (NewProduct):
    description ← NewProduct.description
    entities ← NER (description)
    return entities
```

B. Item Similarity

After obtaining the entity extraction results in the form of E_N, the system proceeds to the item similarity measurement stage. Each E_N representation generated from the NER process on the product will be converted into an embedding vector using an embedding method that is appropriate for the data characteristics, whether using BERT or spaCy.

Similarity measurement is performed using the cosine similarity method, which calculates the similarity angle between two embedding vectors in a high-dimensional space [13]. This score reflects the extent to which two products share similar meanings or attributes. If v_N is the embedding vector of the new product N and v_{C_j} is the embedding vector of product C_j in the database, then the similarity score is calculated using Eq. (1).

$$\text{sim}\left(N, C_j\right) = \frac{v_N \cdot v_{C_j}}{\|v_N\| \times \left\|v_{C_j}\right\|} \quad (1)$$

The result of this process is a list of similarity scores between the new product and all products in the database. The system then filters products with scores above a certain similarity threshold (0.70) [31], forming a set of reference products R that are considered similar.

However, if the set R is empty (no products in the database have a similarity score exceeding the threshold), the system will automatically move to another available database and repeat the similarity search process using the same procedure. This process continues until all databases have been accessed and compared. If, after searching the entire database, the system still cannot find a product that meets the similarity threshold, it will activate a fallback strategy, which is an alternative mechanism that uses a clustering-based approach to group products with similar metadata characteristics (Algorithm 2).

Algorithm 2: Calculating Similarity Score

```
Input: NewEntities, ListProductDatabases
Output: SimilarProducts or FallbackRating
Function Similarity (NewEntities, ListProductDatabases):
    SimilarProducts ← []
    NewEmbed ← BERT (NewEntities)
    For each Database in ListProductDatabases:
        For each Product in Database:
            ExistingEntities ← Product.Entities
            OldEmbed ← Product.Embedding
            score ← CosineSimilarity (NewEmbed, OldEmbed)
            If score ⩾ Threshold:
                SimilarProducts.append({Product, score})
        If SimilarProducts is not empty:
            return SimilarProducts
    FallbackRating ← FallbackClustering, ListProductDatabases)
    return FallbackRating
```

C. Ratings Prediction

The final stage of the model is predicting ratings for new products. After finding a number of similar reference products, the system calculates the estimated rating based on the weighted average of the ratings of those reference products [32] (Algorithm 3). The higher the similarity of a reference product to the new product, the greater its contribution to the rating estimate. The rating estimate for the new product is calculated using Eq. (2).

$$\hat{r}_N = \frac{\sum_{C_j \in R} \text{sim}(N, C_j) \cdot r_{C_j}}{\sum_{C_j \in R} \text{sim}\left(N, C_j\right)} \quad (2)$$

Algorithm 3: Calculating Rating Predictions

```
Input: SimilarProductsList, ProductRatings
Output: PredictedRating
Function PredictRating (SimilarProductsList):
    totalWeight ← 0
    weightedSum ← 0
    For each item in SimilarProductsList:
        Product ← item.Product
        Score ← item.SimilarityScore
        Rating ← Product.Rating
        weightedSum ← weightedSum + (Score × Rating)
        totalWeight ← totalWeight + Score
    PredictedRating ← weightedSum / totalWeight
    return PredictedRating
```

V. Performance Assessment

The system was evaluated to measure the effectiveness of the proposed approach in addressing the cold start problem for new items. The assessment was conducted using two

approaches: quantitative evaluation (based on classification metrics) and qualitative evaluation (through expert validation).

A. Precision, Recall, and F1-Score

System performance was evaluated using three main metrics, namely precision, recall, and F1-score [33]. These three metrics were used to assess the effectiveness of the Named Entity Recognition (NER) model in extracting descriptive entities from new items. The model was trained for 15 epochs with a precision value of 0.838, recall of 0.846, F1-score of 0.842, and accuracy of 0.967.

These results indicate that the system is capable of extracting important entities (products, ingredients, origin) with high precision and completeness, which directly impacts the quality of embedding formation and the accuracy of similarity estimation between items.

As a comparison, the conventional item-item collaborative filtering method based on cosine similarity and Pearson correlation coefficient, adapted from Hasan et al. [34], was reimplemented using the same dataset as this study. Comparison of evaluation matrices can be seen in Table III.

TABLE III. EVALUATION METRICS FOR THE PROPOSED METHOD AND THE CONVENTIONAL METHOD

Metric	Proposed Method	Conventional Method	Improvement (%)
Precision	0.838	0.725	+15.59%
Recall	0.846	0.720	+17.50%
F1-score	0.842	0.722	+16.54%

Based on the results in Table III, it shows that the proposed method outperforms the conventional method in all metrics, with improvements of 15.59% in precision, 17.50% in recall, and 16.54% in F1-score. The highest improvement in recall demonstrates the system's ability to recognise more relevant entities while maintaining prediction accuracy.

B. Training and Validation

The training and validation processes were conducted over 15 epochs on the Named Entity Recognition (NER) model using a dataset divided into two parts: training data for updating model weights and validation data for measuring generalisation ability.

At the start of training, the training loss began at 0.7015 in the first epoch and decreased significantly to 0.0005 in epochs 14 and 15. This pattern indicates that the model quickly learned patterns from the training data. Meanwhile, the validation loss also decreases significantly from 0.4736 to around 0.1367 at the end of training, indicating improved generalisation without significant signs of overfitting.

Overall, this loss value trend indicates that the training process successfully improved the model's performance significantly while maintaining its generalization ability on new data. This suggests that the choice of training parameters, such as learning rate, number of epochs, and batch size, as well as the model architecture used, are appropriate for learning the patterns of entities present in the dataset. As a result, the trained NER model can be relied upon to accurately extract entities from the available product description data. A more detailed visualization of the training and validation loss trends can be seen in Fig. 4.

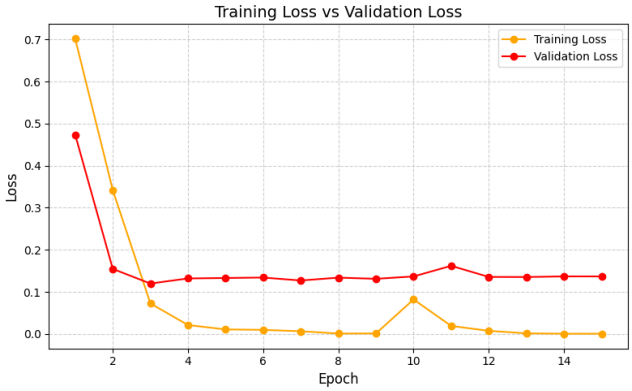


Fig. 4. Training loss and validation loss.

C. Mean Absolute Error (MAE)

Mean Absolute Error (MAE) measurement is used to evaluate the accuracy of rating predictions on new items. A lower MAE value indicates predictions that are closer to the actual rating.

A comparison was made between the proposed method, the method of Kadyanan et al. [35] which combines ICHM, Slope One, and SAW, and the method of Liu et al. [36] which applies an adaptive meta-learning approach with an attention mechanism (AMeLU). Testing using the same dataset yielded the following results in Table IV.

TABLE IV. COMPARISON OF MEAN ABSOLUTE ERROR (MAE)

Method	Lowest MAE	Highest MAE
Proposed Method	0.008	0.319
Kadyanan et al. (2019)	0.110	1.060
Liu et al. (2023)	0.75	0.90

Table IV shows a comparison of the Mean Absolute Error (MAE) values between the proposed method, the method of Kadyanan et al. [35], and Liu et al. [36]. The results show that the method of Kadyanan et al. obtained the lowest MAE value of 0.110, while the proposed method achieved 0.008, indicating a significant improvement in prediction accuracy. On the other hand, the highest MAE value for the Kadyanan et al. method reached 1.060, whereas the proposed method only reached 0.319, demonstrating better stability across scenarios. Furthermore, Liu et al. maintained MAE values within the range of 0.75–0.90, which is more stable than Kadyanan et al. but still less accurate compared to the proposed method. These findings confirm that the proposed method is capable of minimising prediction errors more effectively, especially in cold-start scenarios, thereby producing more consistent and accurate estimates than the benchmark methods.

D. Expert Validation

To complement the quantitative evaluation, manual validation (expert validation) was performed by two experts namely Validator 1 and Validator 2 in the field of

recommendation systems and natural language processing [37]. This validation aimed to assess the extent to which the system was able to generate descriptions of new products that matched products with a history of user interaction. The validation process was carried out by comparing the system's similarity results with the assessments of the two validators, as shown in Table V [38].

TABLE V. VALIDATOR ASSESSMENT CONTINGENCY

	Validator 2 (Valid)	Validator 2 (Invalid)	Total
Validator 1 (Valid)	910	30	940
Validator 1 (Invalid)	10	50	60
Total	920	80	1000

Table V shows the results of the contingency assessment validator, which compares the assessments of two validators on the output of the recommendation system. Symbol a represents the number of data points deemed valid by both validators, totaling 910 data points. b refers to data points deemed valid by Validator 1 but invalid by Validator 2, totaling 30 data points. c denotes data points deemed invalid by Validator 1 but valid by Validator 2, totaling 10 data points. while d represents data deemed invalid by both validators, totaling 50 data points. Thus, the total number of validated data points is T = 1000. Consequently, the agreement between the two validators is represented by the values of a and d, while the disagreement is indicated by the values of b and c. To measure the level of agreement among validators, Cohen's Kappa is used as shown in Eq. (3), (4), and (5).

Actual agreement proportion (P_O):

$$P_O = \frac{a+d}{T} \tag{3}$$

Proportion of chance agreement (P_e):

$$P_e = \left(\frac{a+b}{T} \cdot \frac{a+c}{T}\right) + \left(\frac{c+d}{T} \cdot \frac{b+d}{T}\right) \tag{4}$$

Kappa (K):

$$K = \frac{P_O - P_e}{1 - P_e} \tag{5}$$

Based on the data in Table III, $P_O = 0.96$, $P_e = 0.869$, and K = 0.69 were obtained. According to the interpretation by Madadizadeh et al. [39], Kappa values can be categorized as follows: < 0.00 = Poor agreement, 0.00–0.20 = Slight agreement, 0.21–0.40 = Fair agreement, 0.41–0.60 = Moderate agreement, 0.61–0.80 = Substantial agreement, and 0.81–1.00 = Almost perfect agreement. Thus, a Kappa value of 0.69 falls into the substantial agreement category, indicating a high level of agreement between the two validators. This result confirms that the recommendations generated by the system are deemed consistent and relevant by the experts, thereby strengthening the model's reliability in addressing the cold start problem.

VI. CONCLUSION

This study proposes a metadata-based recommendation system model using a Natural Language Processing (NLP) approach, specifically Named Entity Recognition (NER), to address the cold start problem, particularly in traditional handicraft products. The system extracts descriptive entities such as product type, materials, and regional origin from unstructured text, then converts them into vector representations using adaptive embedding methods. The model integrates a similarity estimation mechanism with a clustering-based fallback strategy to ensure rating predictions can still be made even if no similar items are found in the database.

Experimental results show that the NER model achieves excellent performance with precision of 0.838, recall of 0.846, F1-score of 0.842, and accuracy of 0.967, reflecting the system's ability to accurately extract important entities. Compared to conventional filtering approaches, this model demonstrates significant improvements across all evaluation metrics. Additionally, the system's prediction performance was measured using Mean Absolute Error (MAE), with an average of 0.159, a minimum value of 0.008, and a maximum value of 0.319. The relatively low MAE value indicates that the system's rating estimates are very close to the actual values, thereby reinforcing the model's reliability in the context of recommendations. Validation conducted by two experts showed a substantial agreement level, indicating that the recommendations generated by the system are considered consistent and relevant, especially when dealing with new items (cold start).

Scientifically, this research proves that integrating metadata with NER can be an effective alternative in overcoming the cold start problem, while expanding the application of content-based methods in the local cultural domain. From a practical perspective, this model has the potential to support SMEs and local marketplaces in enhancing the visibility of craft products and promoting cultural initiatives in the digital realm.

This study still has several limitations that can be explored in future research. The recommendation system can be extended by integrating both explicit and implicit user feedback, enabling it to address the cold start problem not only at the item level but also for new users. In addition, the results of Named Entity Recognition (NER) can be enriched with semantic approaches, such as knowledge graphs or semantic embeddings, to better capture contextual meaning and relationships among entities. The proposed framework also has potential to be applied across other product domains, such as fashion, culinary, and tourism, to evaluate scalability and ensure robustness in broader and multilingual contexts. Furthermore, the fallback clustering mechanism can be further developed to handle cases where products have no similarity values above the defined threshold with items in the database. In such scenarios, the system can still provide relevant recommendations by employing alternative clustering approaches, such as Graph-based Latent Allocation (GLA), Incremental Clustering using Hierarchical Merging (ICHM), K-Means Variants, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), or Self-Organizing Maps (SOM).

ACKNOWLEDGMENT

The authors declare that there is no conflict of interest regarding the publication of this article.

REFERENCES

[1] L. Mohanty, L. Saraswat, P. Garg, and S. Lamba, "Recommender Systems in E-Commerce," Proc. - 2022 5th Int. Conf. Comput. Intell. Commun. Technol. CCICT 2022, no. July, pp. 114–119, 2022, doi: 10.1109/CCiCT56684.2022.00032.

[2] Vrushal Jagtap, Om Mahajan, Bhavesh Rane, Kumar Kolhe, and Priyanka Bendale, "General Purpose Recommendation System," Int. J. Adv. Res. Sci. Commun. Technol., vol. 3, no. 1, pp. 278–280, 2023, doi: 10.48175/ijarsct-9040.

[3] R. S. Sharma, A. A. Shaikh, and E. Li, "Designing Recommendation or Suggestion Systems: looking to the future," Electron. Mark., vol. 31, no. 2, pp. 243–252, 2021, doi: 10.1007/s12525-021-00478-z.

[4] M. Ibrahim, I. S. Bajwa, N. Sarwar, F. Hajjej, and H. A. Sakr, "An Intelligent Hybrid Neural Collaborative Filtering Approach for True Recommendations," IEEE Access, vol. 11, no. June, pp. 64831–64849, 2023, doi: 10.1109/ACCESS.2023.3289751.

[5] R. Khatwal and S. N. Hasan, "Cold Start Problem in Recommendation System: A Solution Model Based on Clustering and Association Rule Techniques," 2022 5th Int. Conf. Multimedia, Signal Process. Commun. Technol. IMPACT 2022, pp. 1–8, 2022, doi: 10.1109/IMPACT55510.2022.10029293.

[6] N. AlRossais, D. Kudenko, and T. Yuan, Improving cold-start recommendations using item-based stereotypes, vol. 31, no. 5. Springer Netherlands, 2021. doi: 10.1007/s11257-021-09293-9.

[7] J. Wei, J. He, K. Chen, Y. Zhou, and Z. Tang, "Collaborative filtering and deep learning based recommendation system for cold start items," Expert Syst. Appl., vol. 69, pp. 1339–1351, 2017, doi: 10.1016/j.eswa.2016.09.040.

[8] A. Gharahighehi, K. Pliakos, and C. Vens, "Addressing the Cold-Start Problem in Collaborative Filtering Through Positive-Unlabeled Learning and Multi-Target Prediction," IEEE Access, vol. 10, no. November, pp. 117189–117198, 2022, doi: 10.1109/ACCESS.2022.3219071.

[9] R. Widayanti, M. H. R. Chakim, C. Lukita, U. Rahardja, and N. Lutfiani, "Improving Recommender Systems using Hybrid Techniques of Collaborative Filtering and Content-Based Filtering," J. Appl. Data Sci., vol. 4, no. 3, pp. 289–302, 2023, doi: 10.47738/jads.v4i3.115.

[10] W. Shalaby, S. Oh, A. Afsharinejad, S. Kumar, and X. Cui, M2TRec: Metadata-aware Multi-task Transformer for Large-scale and Cold-start free Session-based Recommendations, vol. 1, no. 1. Association for Computing Machinery, 2022. doi: 10.1145/3523227.3551477.

[11] K. N. Mishra, A. Mishra, P. N. Barwal, and R. K. Lal, "Natural Language Processing and Machine Learning-Based Solution of Cold Start Problem Using Collaborative Filtering Approach," Electron., vol. 13, no. 21, 2024, doi: 10.3390/electronics13214331.

[12] J. Barroca, A. Shivkumar, B. Q. Ferreira, E. Sherkhonov, and J. Faria, "Enriching a Fashion Knowledge Graph from Product Textual Descriptions," CEUR Workshop Proc., vol. 3184, pp. 36–51, 2022.

[13] J. Zhang, F. Wang, F. Ma, and G. Song, "Text Similarity Calculation Method Based on Optimized Cosine Distance," Proc. - 2022 Int. Conf. Electron. Devices, Comput. Sci. ICEDCS 2022, pp. 37–39, 2022, doi: 10.1109/ICEDCS57360.2022.00015.

[14] C. Xiao et al., "UPRec: User-aware Pre-training for sequential Recommendation," AI Open, vol. 4, no. September, pp. 137–144, 2023, doi: 10.1016/j.aiopen.2023.08.008.

[15] H. Zhou, F. Xiong, and H. Chen, "A Comprehensive Survey of Recommender Systems Based on Deep Learning," Appl. Sci., vol. 13, no. 20, 2023, doi: 10.3390/app132011378.

[16] L. Jiang, L. Liu, J. Yao, and L. Shi, "A hybrid recommendation model in social media based on deep emotion analysis and multi-source view fusion," J. Cloud Comput., vol. 9, no. 1, 2020, doi: 10.1186/s13677-020-00199-2.

[17] E. S. Negara, Sulaiman, R. Andryani, P. H. Saksono, and Y. Widyanti, "Recommendation System with Content-Based Filtering in NFT Marketplace," J. Adv. Inf. Technol., vol. 14, no. 3, pp. 518–522, 2023, doi: 10.12720/jait.14.3.518-522.

[18] S. Lestari, Yulmaini, Aswin, S. Y. Ma'ruf, Sulyono, and R. R. N. Fikri, "Alleviating cold start and sparsity problems in the micro, small, and medium enterprises marketplace using clustering and imputation techniques," Int. J. Electr. Comput. Eng., vol. 14, no. 3, pp. 3220–3229, 2024, doi: 10.11591/ijece.v14i3.pp3220-3229.

[19] A. Panteli and B. Boutsinas, "Addressing the Cold-Start Problem in Recommender Systems Based on Frequent Patterns," Algorithms, vol. 16, no. 4, 2023, doi: 10.3390/a16040182.

[20] P. Gupta, A. Deo, S. S. Khan, A. S. Rajput, K. Joshi, and A. Chourasia, "A Progressive Approach for a Personalized Recommendation System Using Hybrid Deep Learning," Ing. des Syst. d'Information, vol. 29, no. 5, pp. 1999–2009, 2024, doi: 10.18280/isi.290531.

[21] M. T. Ramakrishna et al., "HCoF: Hybrid Collaborative Filtering Using Social and Semantic Suggestions for Friend Recommendation," Electron., vol. 12, no. 6, pp. 1–23, 2023, doi: 10.3390/electronics12061365.

[22] Y. Afoudi, M. Lazaar, and M. Al Achhab, "Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network," Simul. Model. Pract. Theory, vol. 113, no. April, p. 102375, 2021, doi: 10.1016/j.simpat.2021.102375.

[23] A. Vomberg, N. Schauerte, S. Krakowski, C. Ingram Bogusz, M. J. Gijsenberg, and A. Bleier, "The cold-start problem in nascent AI strategy: Kickstarting data network effects," J. Bus. Res., vol. 168, no. December 2022, p. 114236, 2023, doi: 10.1016/j.jbusres.2023.114236.

[24] Z. P. Zhang, Y. Kudo, T. Murai, and Y. G. Ren, "Addressing complete new item cold-start recommendation: A niche item-based collaborative filtering via interrelationship mining," Appl. Sci., vol. 9, no. 9, 2019, doi: 10.3390/app9091894.

[25] X. Shen, X. Hu, N. Liu, and Y. Shen, "Named Entity Recognition of Multilevel Features," 2022 4th Int. Conf. Front. Technol. Inf. Comput. ICFTIC 2022, pp. 575–578, 2022, doi: 10.1109/ICFTIC57696.2022.10075239.

[26] P. Chaudhry, "Bidirectional Encoder Representations from Transformers for Modelling Stock Prices," Int. J. Res. Appl. Sci. Eng. Technol., vol. 10, no. 2, pp. 896–901, 2022, doi: 10.22214/ijraset.2022.40406.

[27] B. B. Elov, "Making Use of a 'Spacy' Module in the Natural Language Processing," J. Sci. Innov. Dev., vol. 5, no. 4, pp. 41–54, 2022, doi: 10.36522/2181-9637-2022-4-5.

[28] A. Senior, "Pearson ' s and Cosine Correlation," Int. Conf. Trends Electron. Informatics ICEI 2017 Calc., vol. 05 June 20, no. May, pp. 1000–1004, 2017.

[29] M. Ghiasabadi Farahani, J. Akbari Torkestani, and M. Rahmani, "Adaptive personalized recommender system using learning automata and items clustering," Inf. Syst., vol. 106, p. 101978, 2022, doi: 10.1016/j.is.2021.101978.

[30] M. A. Al-Hagery, "A novel Based-Approach Composed of Clustering Algorithm & Cosine Similarity for Products Recommendation," Int. J. Educ. Inf. Technol., vol. 14, pp. 133–141, 2020, doi: 10.46300/9109.2020.14.16.

[31] M. D. Khilji, "Features Matching using Natural Language Processing," Int. J. Cybern. Informatics, vol. 12, no. 2, pp. 251–260, 2023, doi: 10.5121/ijci.2023.120218.

[32] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," Expert Syst. Appl., vol. 41, no. 4 PART 2, pp. 2065–2073, 2014, doi: 10.1016/j.eswa.2013.09.005.

[33] S. Al Rawahi, N. Al Musalhi, and E. Celebi, "Recommendation Systemfor Daily Consumer Purchases List Usingspecial Hybrid Approach," vol. 30, no. 4, pp. 1239–1250, 2024, doi: 10.53555/kuey.v30i4.1642.

[34] M. Hasan and F. Roy, "An item–item collaborative filtering recommender system using trust and genre to address the cold-start problem," Big Data Cogn. Comput., vol. 3, no. 3, pp. 1–15, 2019, doi: 10.3390/bdcc3030039.

[35] I. G. A. G. A. Kadyanan, I. B. G. Dwidasmara, I. B. M. Mahendra, I. K. Ari Mogi, and I. W. Puguh Sudarma, "The design of typical Balinese food recommendation system using hybrid method of collaborative filtering and Slope One algorithm on mobile device platform," ACM Int. Conf. Proceeding Ser., pp. 111–116, 2019.

[36] S. Liu, Y. Liu, X. Zhang, C. Xu, J. He, and Y. Qi, "Improving the Performance of Cold-Start Recommendation by Fusion of Attention

Network and Meta-Learning," Electron., vol. 12, no. 2, pp. 1–14, 2023, doi: 10.3390/electronics12020376.

[37] W. Al Jurdi, J. B. Abdo, J. Demerjian, and A. Makhoul, "Group Validation in Recommender Systems: Framework for Multi-layer Performance Evaluation," ACM Trans. Recomm. Syst., vol. 2, no. 1, pp. 1–25, 2024, doi: 10.1145/3640820.

[38] K. S. Tan, Y. C. Yeh, P. S. Adusumilli, and W. D. Travis, "Quantifying Interrater Agreement and Reliability Between Thoracic Pathologists: Paradoxical Behavior of Cohen's Kappa in the Presence of a High Prevalence of the Histopathologic Feature in Lung Cancer," JTO Clin. Res. Reports, vol. 5, no. 1, p. 100618, 2024, doi: 10.1016/j.jtocrr.2023.100618.

[39] F. Madadizadeh, H. Ghafari, and S. Bahariniya, "The Open Public Health Journal Kappa Statistics: A Method of Measuring Agreement in Dental Examinations," Open Public Health J., vol. 16, pp. 1–6, 2023, doi: 10.2174/0118749445259818231016040344.

Analysis of Factors Affecting Continuance Intention in Indonesian Digital Banks

Soros Lie, Viany Utami Tjhin
Information Systems Department-School of Information Systems, Bina Nusantara University, Jakarta, Indonesia 11480

***Abstract*—Indonesian Digital Banks are currently competing to get more customers with their own mobile application, digital finance ecosystem and their promotion method. The research aims to find out the factors that influence customer satisfaction when using the digital bank application. The variables used in this study are System Quality, Service Quality, Information Quality, Perceived Advantage Effort Expectancy, Digital Ecosystem, User Satisfaction, and Continuance Intention in the mobile application. The results of the study concluded what factors influence satisfaction to develop a better mobile application for an Indonesian digital bank. The research model and questionnaire use a modified research model of the successful information system model DeLone and McLane and in analyzing the results of the questionnaire, researchers used descriptive statistics and Structural Equation Model (SEM) analysis using SMART PLS V.4. The results of the management of these data conducted that there are 11 significant direct effects on User Satisfaction or Continuance Intention, and 6 direct effects that were not significant. There were also 8 significant indirect effects, mainly where User Satisfaction mediated the impact of other variables on Continuance Intention. The results of this study are expected to provide a better reference for Indonesian digital banks to improve their mobile app services and maintain customer loyalty.**

***Keywords*—Digital banking; DeLone and McLean; structural equation model; continuance intention; user satisfaction**

I. Introduction

The growth of information technology in Indonesia is continuing to accelerate and has significantly influenced the country's banking sector. Based on the results of the Sharing Vision Survey, mobile banking recorded the sharpest increase, rising from 34% in 2021 to 56% in 2022 [1]. This percentage was higher than the combined total of the other four channels. Additionally, users preferred mobile banking for checking transaction history, checking balances, transferring funds, and paying bills compared to other methods. This growth, coupled with digital banks' ability to lower costs, connects underserved populations to formal financial services [2]. These developments highlight the growing importance of digital banking and provide the context for examining what drives users to keep engaging with these applications over time.

Digital banks and conventional banks have key differences in operational models, functionality, services, and challenges. Digital banks primarily operate through electronic channels without physical branches [3]. In Indonesia, the regulation of digital banks and conventional banks is governed by regulations issued by Indonesia's Financial Services Authority (OJK). POJK (Financial Services Authority Regulation) No.12/POJK.03/2021. By understanding these structural and regulatory differences, this study situates digital banking as a technology-reliant service environment where user experience becomes a key determinant of success.

Competition among digital companies has intensified due to the rapid emergence of new digital banks, while conventional banks have also begun developing digital offerings. Digital banks attract users with a variety of features and a simpler, faster registration process compared to conventional banks, which often require more complex procedures. Among the leading players, SeaBank and Bank Jago have the largest user bases and the highest ratings on both the App Store and Google Play Store. SeaBank holds a rating of 4.9 on both platforms [4][5], while Jago scores 4.5 on the App Store and 4.9 on Google Play [6][7]. Despite these high ratings, both banks still receive a substantial volume of user complaints, especially regarding application performance and stability, login and authentication issues, and transaction errors. Additional concerns include user interface challenges, customer service responsiveness, and system downtime. These recurring problems reveal a persistent gap between user expectations and actual system performance, which directly affects satisfaction and long-term loyalty. Customer satisfaction has been widely recognized as a crucial determinant of system success and continuance intention in information systems research [8]. The original DeLone & McLean IS Success Model has been widely applied to evaluate information system success through the dimensions of System Quality, Information Quality, and Service Quality. Despite its broad adoption, several limitations have been highlighted in the context of digital banking. The model was conceptualized for single-system environments and does not fully capture the value generated by platform ecosystems, which now play a central role as banks compete as orchestrators of partner services and cross-industry integrations [9]. Moreover, the model does not explicitly account for the user's benefit–effort calculus, such as the perceived advantage of using digital banking over alternative services and the perceived effort required to interact with the system. Insights from UTAUT-based studies demonstrate that these factors are strong predictors of continuance intention and complement, rather than duplicate, the traditional quality dimensions [10]. Empirical findings on the DeLone & McLean relationships are mixed, suggesting additional constructs are needed. These limitations provide the rationale for extending the model with Digital Ecosystem, Perceived Advantage, and Effort Expectancy to better capture the drivers of satisfaction and long-term use in Indonesia's digital banking sector.

The remainder of this paper is organized as follows: Section II reviews the literature and presents the research model. Section III describes the methodology. Section IV reports the

SEM results, Section V discusses implications, and Section VI concludes with recommendations and directions for future research.

A. Research Questions

In view of the research problem, this research seeks to address the following research questions:

- What factors that affects the continuance intention and customer satisfaction in Indonesian's digital banks?

B. Research Objective

This research focuses on digital banks in Indonesia, particularly in relation to customer satisfaction and continuance intention. The objects of the study are customers who use the top two Indonesian digital bank applications—PT Bank Seabank Indonesia and PT Bank Jago Tbk which have the highest ratings in Google Play and the App Store, as well as the largest user bases. The data collection method used in this research is an online questionnaire distributed via social media. The target number of respondents is 400 (four hundred), determined using the Slovin formula to obtain a valid sample size.

II. Literature Review

A. Digital Bank

A digital bank is a Bank with Indonesian legal entities (Bank BHI) that operates primarily through electronic channels without physical branches except for the head office, or it uses a limited number of physical offices. Digital banks can be established either by setting up a new Bank BHI specifically as a digital bank or by transforming an existing Bank BHI into a digital bank. The regulation categorizes digital banks under the framework of Bank BHI, which allows them to operate with a unique model of minimal physical presence, relying instead on electronic means to serve customers. This regulatory approach is designed to foster innovation in the banking sector, accommodating evolving consumer preferences towards digital banking solutions. This model supports the banking industry's drive towards enhancing efficiency, reducing costs, and expanding accessibility, aligning with the broader goals of increasing competitiveness and contributing to the national economy as outlined by the OJK [11].

B. System Quality, Service Quality and Information Quality

In the original DeLone and McLean IS Success Model, the critical dimensions of System Quality, Service Quality, and Information Quality are integral to evaluating the success of information systems. System Quality focuses on the usability, responsiveness, flexibility, and reliability of the system, which are essential for ensuring effective user interaction and satisfaction, Information Quality enhances decision-making and user satisfaction through its emphasis on the timeliness, completeness, relevance, and accuracy of information provided and , Service Quality reflects the efficiency and responsiveness of support services, which are crucial for maintaining user trust and encouraging continuous use of the system [12]. Together, these dimensions form the core of the DeLone and McLean model, highlighting the importance of quality in various facets of digital banking systems to achieve overall system success.

C. Perceived Advantage

Perceived advantage refers to the perceived relative advantage of an information system by the users in terms of information content, system interface, and functional services compared with other types of information systems [10]. This includes elements such as promotions, which can significantly impact a user's intention to use a digital financial product or service. Previous research indicates that these perceived advantages have a positive impact on user's intention to use and their satisfaction with the provided service [9].

D. Effort Expectancy

Effort expectancy is a critical factor in understanding the adoption and continuous use of electronic banking services. According to [13], effort expectancy involves the perceived ease associated with using electronic banking systems. It highlights that effort expectancy significantly influences customer usage of digital banking services. The research findings suggest that when customers perceive less effort required to use these services, their actual usage increases. This relationship is noted to be having strong importance of reducing perceived effort to enhance customer adoption rates. Improving consumers' perceptions of how easy electronic banking services are to use should be a priority for banks aiming to expand their digital service utilization

E. User Satisfaction

User satisfaction is defined by [14] as the degree of ease and approval users experience during their interactions with a system through an application. This aspect is crucial because it directly influences the overall success of digital banking applications. Satisfying user experiences foster higher engagement rates, as users are more likely to continue using services that meet or exceed their expectations. Furthermore, user satisfaction is a key determinant of customer loyalty and can significantly affect word-of-mouth recommendations, which are vital for the organic growth of any service. Positive user experiences that lead to high satisfaction levels also reduce the churn rate; whereby fewer users discontinue using the service due to dissatisfaction [14]. Therefore, focusing on enhancing user satisfaction is not only about ensuring that users are pleased in the short term but also about securing their long-term commitment and advocacy.

F. Continuance Intention

Continuance intention reflects users' commitment to keep using digital banking applications regularly, avoid switching to alternatives, and even recommend them to others [14]. In Indonesia, this intention grows stronger when users feel that digital banking offers easy access, secure transactions, and practical benefits for daily financial needs, which reduces the chance of moving to other application. Other research also shows that satisfaction and the match between user expectations and actual experience play an important role, because when the app runs smoothly and reliably, people are more likely to keep using it in the long run [15].

G. Partial Least Square Structural Equation Modeling

Memon et al. [16] explain that Partial Least Squares Structural Equation Modeling (PLS-SEM) is particularly advantageous for prediction-oriented studies, exploratory

research, and theory development, especially when datasets do not satisfy the strict assumptions of covariance-based SEM. This method is capable of handling complex models involving latent constructs and effectively addresses measurement error, which is inherent in the evaluation of abstract variables. PLS-SEM offers flexibility for researchers to simultaneously evaluate measurement and structural models, making it a robust method for this research [16].

H. Extended DeLone and McLean's IS (Information Systems) Success Model

Prior studies confirm that System Quality, Information Quality, and Service Quality significantly influence user satisfaction and trust, although results vary by context [17]. Research on Digital Ecosystem emphasizes that ecosystem breadth, partner integration, and seamless cross-platform experiences are critical drivers of satisfaction and engagement in digital financial services [9]. Effort Expectancy has been consistently linked to behavioral intention in UTAUT-based models, reflecting the importance of ease of use and low cognitive burden for continuance [18]. Perceived Advantage capturing relative benefit, convenience, and cost savings has been found to be a direct and significant predictor of adoption and continued use of mobile banking services [10]. Together, these studies support the inclusion of DE, EE, and PA as extensions to the Delone & McLean model to better explain user behavior in the competitive Indonesian digital banking landscape.

III. Methodology

A. Research Design

This study begins by explaining the general background of Indonesia's digital bank mobile applications, which have been increasingly and massively adopted by Indonesian citizens as part of their daily financial activities [1]. After identifying the problem, a literature study is carried to look for all the supporting studies related to this research topic, followed by the formulation of research questions. To collect primary data, questionnaires were distributed to randomly selected respondents who met the predetermined criteria. After the minimum sample requirements were fulfilled, the data were processed and analyzed.

After the analysis has been conducted, it will also include results that could conclude the research to be used as suggestions or as references for either future research or to improve Indonesia's current digital banking application.

B. Determine Sample

Based on data obtained from Infobanknews, PT Bank Seabank Indonesia has a total of 13.000.000 users [19], while PT Bank Jago Tbk has 12.500.000 users [20], resulting in a combined total of 25,500,000 users.

Sampling is essential in research as it is often impractical to study an entire population, making proper sampling methods critical for valid results [21]. The sample size was determined using the Slovin formula, and it is formally expressed as follows:

$$n = \frac{N}{1 + N\,e^2}$$

Where:

n = Sample

N = Total Population

e = Estimated error rate

Then:

$$n = \frac{13{,}000{,}000 + 12{,}500{,}000}{1 + (13{,}000{,}000 + 12{,}500{,}000)0{,}05^2} = 400.016$$

$$\approx 400\ Sample$$

Based on the proportion of the population, Seabank users represent approximately 50.98% of the total, while Bank Jago users account for 49.02%. Therefore, of the 400 targeted respondents, 204 were allocated for Seabank users and 196 for Bank Jago users. The sample criteria are based on the terms and conditions of PT Bank Seabank Indonesia and PT Bank Jago Tbk, which set the minimum age requirement for customers at 17 years old.

C. Research Model

The study was conducted using a modified DeLone and McLean model, where the results identify the factors that influence Continuance Intention. In this framework, continuance intention is affected both directly by these factors and indirectly through the mediating role of user satisfaction. The research framework is illustrated in Fig. 1:

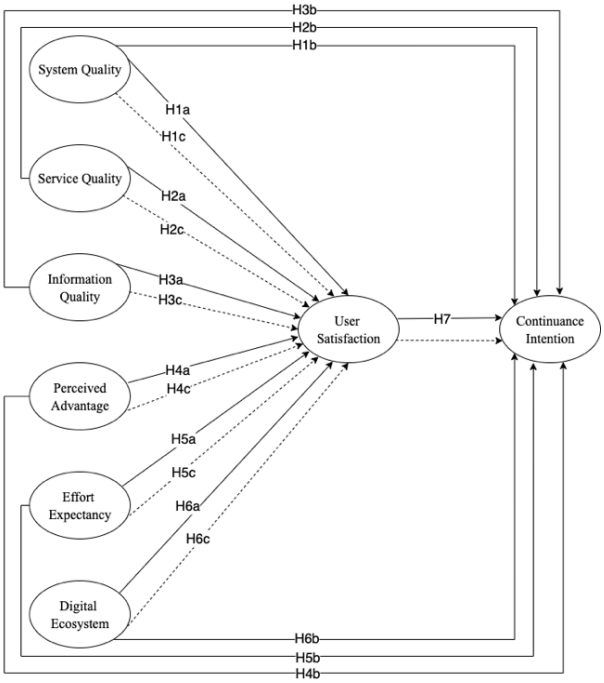


Fig. 1. Research model.

Different from previous research, this research added variables like perceived advantage, effort expectancy and digital ecosystem. The research was conducted using variables by the research model, but each variable that has been determined in the research model has questions from different indicators from previous research.

D. Research Hypothesis

Based on the modification model of DeLone and McLean in Fig. 1. System quality, service quality, information quality, perceived advantage, effort expectancy, and digital ecosystem function as the independent variables. User satisfaction serves as both a dependent variable and a mediating variable, as it is influenced by the independent variables and, in turn, affects continuance intention. Continuance intention is the final dependent variable, representing the outcome that the study seeks to explain. In this research the hypothesis proposed is as follows:

H1a: System quality in Indonesian digital bank applications significantly determines user satisfaction.

H1b: System quality in Indonesian digital bank applications significantly determines continuance intention.

H1c: System quality in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H2a: Service quality in Indonesian digital bank applications significantly determines user satisfaction.

H2b: Service quality in Indonesian digital bank applications significantly determines continuance intention.

H2c: Service quality in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H3a: Information quality in Indonesian digital bank applications significantly determines user satisfaction.

H3b: Information quality in Indonesian digital bank applications significantly determines continuance intention.

H3c: Information quality in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H4a: Perceived advantage in Indonesian digital bank applications significantly determines user satisfaction.

H4b: Perceived advantage in Indonesian digital bank applications significantly determines continuance intention.

H4c: Perceived advantage in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H5a: Effort expectancy in Indonesian digital bank applications significantly determines user satisfaction.

H5b: Effort expectancy in Indonesian digital bank applications significantly determines continuance intention.

H5c: Effort expectancy in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H6a: Digital ecosystem in Indonesian digital bank applications significantly determines user satisfaction.

H6b: Digital ecosystem in Indonesian digital bank applications significantly determines continuance intention.

H6c: Digital ecosystem in Indonesian digital bank applications significantly determines continuance intention through the mediation of user satisfaction.

H7: User satisfaction in Indonesian digital bank applications significantly determines continuance intention.

IV. Data Analysis

From the results of the questionnaire distribution in this study, a total of, 500 valid responses were used for data analysis. This number exceeded the minimum target sample size set for the study, which was 204 respondents for the Seabank application and 196 respondents for the Bank Jago application. In practice, the final dataset consisted of 276 Seabank users and 224 Bank Jago users, thus meeting and surpassing the targeted number of respondents. The data analysis was conducted using SmartPLS V.4 software.

A. Respondent Data Distribution by Respondents by Age

Based on the distributed online survey, respondent data were categorized into four age groups for both Bank Jago and Seabank users, as illustrated in Fig. 2. Most respondents were in the 17–27-year age group, comprising Bank Jago (157) and Seabank (237) users. The second-largest group was aged 28–43 years, with Bank Jago (61) and Seabank (37) users. A smaller portion of respondents were aged 44–58 years, totaling Bank Jago (5) and Seabank (2) users. The smallest age group was respondents aged 59 years and above, consisting of only Bank Jago (1) user and none from Seabank. Judging from this distribution, it can be concluded that digital banking application users in Indonesia are predominantly young adults in the 17–27-year age range, indicating a strong adoption rate of digital banking services among the younger demographic segment.

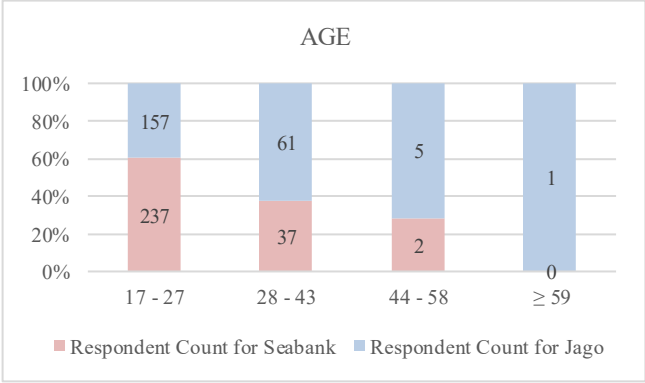


Fig. 2. Proportional distribution of respondent demographics by age.

B. Respondent Data Distribution by Respondents by Gender

Based on the survey results presented in Fig. 3, the gender distribution of respondents for digital banking users in Indonesia shows that 276 individuals (61%) are women, while 176 individuals (39%) are men. In Seabank, 213 respondents were women and 63 were men, whereas in Jago, 168 respondents were women and 56 were men. These results indicate that both men and women actively use digital banking services in Indonesia, suggesting that adoption is not limited by specific gender criteria.

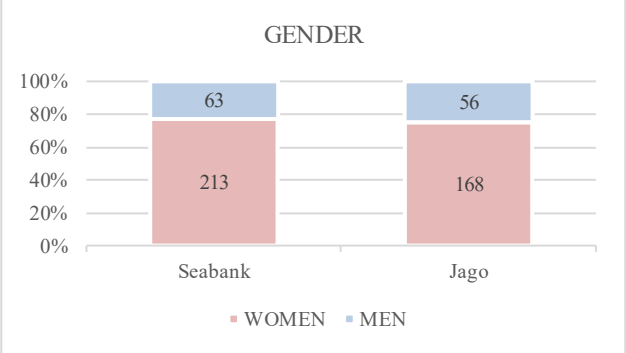


Fig. 3. Proportional distribution of respondent demographics by gender.

C. Respondent Data Distribution by Respondents by Occupation

Based on the collected questionnaire data, respondents' occupations were distributed across seven categories for both Seabank and Jago users, as illustrated in Fig. 4. For Seabank, the largest proportion of respondents were students (174), followed by private employees (75), freelance workers (16), civil servants/ASN (5), entrepreneurs (4), homemakers (1), and informal workers (1). For Jago, the highest number of respondents were also students (124 individuals), followed by private employees (64), freelance workers (16), civil servants/ASN (8), entrepreneurs (8), homemakers (4), and no respondents in the informal worker category. These findings indicate that students and private employees form the dominant user segments for both digital banking applications, suggesting that these demographic groups are particularly engaged with such digital financial technologies.

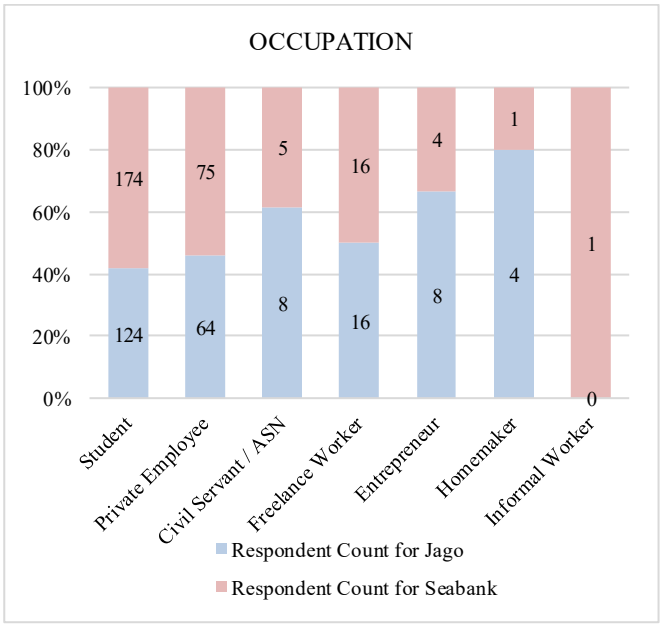


Fig. 4. Proportional distribution of respondent demographics by occupation.

D. Respondent Data Distribution by Respondents by Usage Duration

Based on the survey results presented in Fig. 5, the distribution of respondents by application usage duration for digital banking in Indonesia shows variation across different time periods. For Seabank, the breakdown was: less than 1 month (11), 1–6 months (70), 7–12 months (53), 1–2 years (84), and more than 2 years (58). For Jago, the figures were: less than 1 month (12), 1–6 months (57), 7–12 months (59), 1–2 years (69), and more than 2 years (27).

Notably, the group with usage duration of less than 1 month represents the smallest segment for both banks, indicating that a limited proportion of users are in the very early adoption stage. This suggests that most respondents are already beyond the initial trial phase, with a significant number maintaining usage for a year or more, reflecting relatively strong retention trends in Indonesia's digital banking sector.

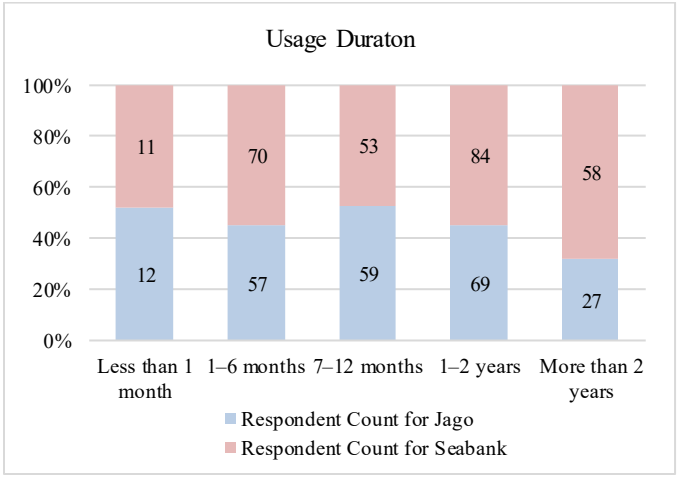


Fig. 5. Proportional distribution of respondent demographics by usage duration.

E. Respondent Data Distribution by Purpose of Using the Application

Based on the survey results presented in Fig. 6, the purposes for using digital banking applications among respondents show a clear preference for certain features. The highest proportion of responses was recorded for promotional incentives (121), highlighting that financial benefits are the most significant factor influencing usage. This was followed by a user-friendly interface (135) and exemption from monthly administrative fees (89), both of which reflect the importance of convenience and cost savings. Other purposes included ease of account opening (44), comprehensive functionality (28), and financial and investment features (26), which indicate interest in broader service capabilities. Security and information-related aspects, such as secure application environment (17) and clear and concise information (14), were less frequently selected. Meanwhile, attractive design (15) and fast, reliable access (11) received the lowest counts.

Overall, these findings suggest that while usability and functionality are valued, the strongest motivator for adopting digital banking applications in Indonesia lies in the financial benefits offered particularly financial related promotional incentives.

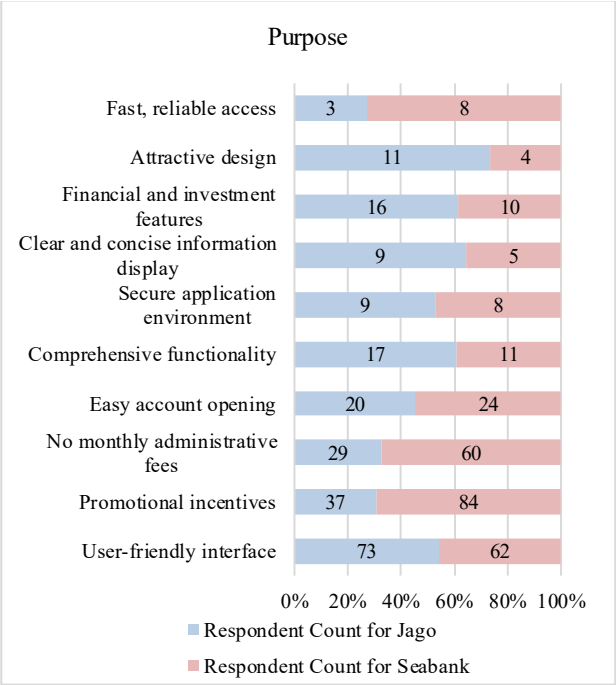


Fig. 6. Proportional distribution of respondent demographics by purpose of application use.

F. Respondent Data Distribution by Usage Frequency

Based on the survey results shown in Fig. 7, the distribution of respondents by application usage frequency for digital banking in Indonesia reflects varied engagement levels. The highest proportion reported using the application once a week (85), followed by every 2–3 days (110) and more than once a day (80). Other usage patterns included once a day (72), every 4–6 days (36), once a month (51), more than a week and less than two weeks (38), and every two weeks (28).

Although a notable share of users accesses the application multiple times per week or more frequently, the data also reveal that a significant portion of users engage less regularly. This lower usage frequency may indicate that some customers still prefer alternative banking channels or payment methods, suggesting both a potential challenge in fostering daily reliance and an opportunity for further growth through increased user retention strategies.

G. Validity Test

This research utilizes questionnaires; it is necessary to conduct a validity test to ensure that the distributed instruments are appropriate and capable of accurately measuring the intended constructs. A valid instrument indicates that the questionnaire items effectively represent the variables under study. In this research, validity testing was performed using SmartPLS v4 based on the preliminary questionnaire tabulation results. The validity of each indicator was evaluated through the factor loading values generated by SmartPLS. An indicator is considered valid if its standardized loading factor exceeds the recommended threshold of 0.70, while items below this cutoff are discarded as they do not sufficiently capture the construct [22]. The results of the validity testing for each research variable are presented in Table I.

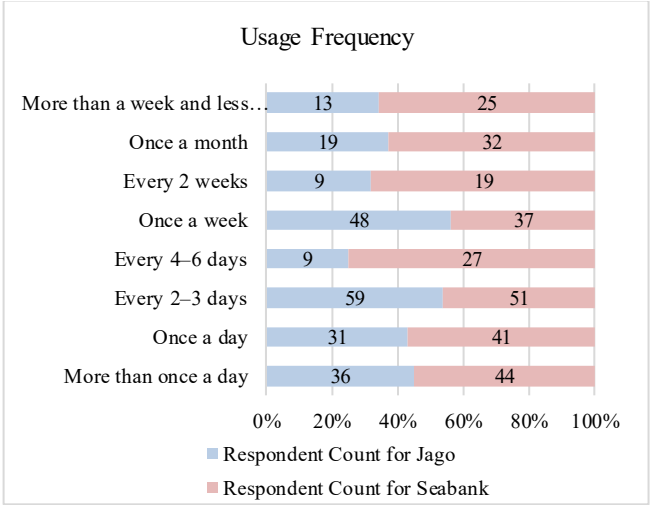


Fig. 7. Proportional distribution of respondent demographics by usage frequency.

TABLE I. VALIDITY TEST

Variable	Question	Result	Explanation
System Quality	SQ1	0.790	Valid
System Quality	SQ2	0.819	Valid
System Quality	SQ3	0.842	Valid
Service Quality	EQ1	0.753	Valid
Service Quality	EQ2	0.637	Invalid
Service Quality	EQ3	0.810	Valid
Service Quality	EQ4	0.738	Valid
Service Quality	EQ5	0.785	Valid
Information Quality	IQ1	0.770	Valid
Information Quality	IQ2	0.733	Valid
Information Quality	IQ3	0.794	Valid
Information Quality	IQ4	0.790	Valid
Information Quality	IQ5	0.752	Valid
Perceived Advantage	PA1	0.742	Valid
Perceived Advantage	PA2	0.804	Valid
Perceived Advantage	PA3	0.746	Valid
Effort Expectancy	EE1	0.826	Valid
Effort Expectancy	EE2	0.803	Valid
Effort Expectancy	EE3	0.839	Valid
Effort Expectancy	EE4	0.787	Valid
Digital Ecosystem	EC1	0.802	Valid
Digital Ecosystem	EC2	0.788	Valid
Digital Ecosystem	EC3	0.779	Valid
Digital Ecosystem	EC4	0.609	Invalid
User Satisfaction	US1	0.841	Valid
User Satisfaction	US2	0.841	Valid
User Satisfaction	US3	0.779	Valid
Continuance Intention	CI1	0.866	Valid
Continuance Intention	CI2	0.846	Valid
Continuance Intention	CI3	0.844	Valid

Based on the results of the SmartPLS analysis shown in Table I, all indicators have loading factor values above the minimum validity threshold of 0.70, with the exception of indicator EQ2 (0.637) and EC4 (0.609), which do not meet the criteria and are therefore considered invalid. Consequently, the remaining indicators that meet or exceed the 0.70 threshold are deemed valid, and the evaluation of the measurement model can be continued without the invalid indicators.

H. Reliability Test

To evaluate the reliability of the constructs, a construct reliability assessment was carried out after the indicator validity test. As described in the previous section, indicators EQ2 (0.637) and EC4 (0.609) did not meet the minimum validity threshold of 0.70 and were therefore removed from the model. The reliability test was then conducted using only the valid indicators that met the threshold.

TABLE II. RATIO RELIABILITY ASSESSMENT

Construct	Result	Explanation
System Quality	0.858	Reliable
Service Quality	0.871	Reliable
Information Quality	0.878	Reliable
Perceived Advantage	0.808	Reliable
Effort Expectancy	0.887	Reliable
Digital Ecosystem	0.860	Reliable
User Satisfaction	0.861	Reliable
Contiuance intention	0.888	Reliable

As shown in Table II, the calculations' results show that all the constructs are reliable as there is no construct whose reliability value is below 0.7, which shows that all the constructs are deserving to be used in the research model.

For reproducibility and clarity, the complete list of variables, codes, and measurement indicators is provided in Table VII (Appendix A). The full wording of all questionnaire items, including demographic and usage-experience questions, is presented in Table VIII (Appendix A).

I. Structural Equation Modeling (SEM)

The research structural model was assessed using SmartPLS v4 to examine the relationships between the exogenous variables System Quality (SQ), Service Quality (EQ), Information Quality (IQ), Perceived Advantage (PA), Effort Expectancy (EE), and Digital Ecosystem (EC) and the endogenous variables User Satisfaction (US) and Continuance Intention (CI).

The evaluation focused on both direct and indirect effects, with path coefficients (β), t-values, and p-values obtained through the bootstrapping procedure (5000 subsamples). In the measurement model assessment, indicators with standardized loading factors below the 0.70 threshold were discarded to improve construct validity and reliability, ensuring that only valid items contributed to the final analysis. Fig. 8 presents the path diagram of the structural model as generated in SmartPLSAnalysis of Direct Effects, Indirect Effects and Total Effects.

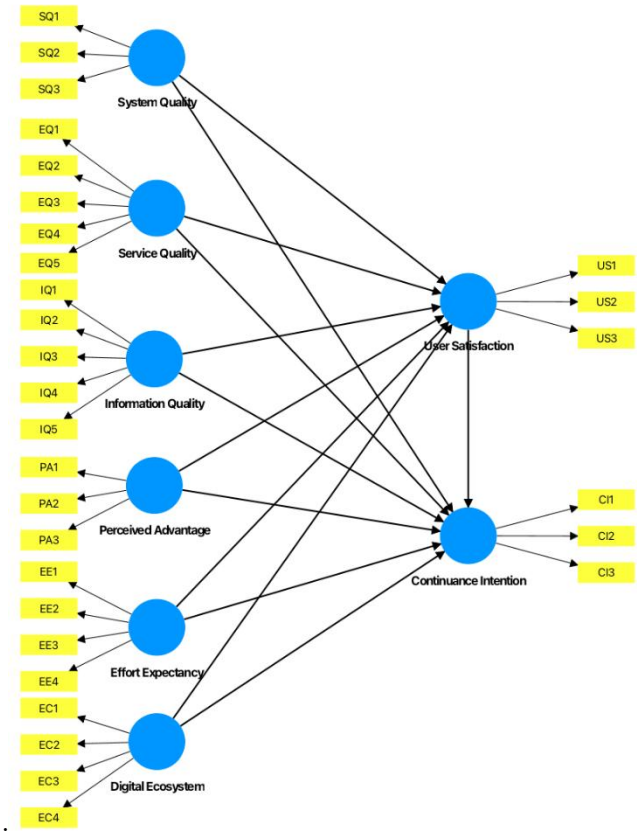


Fig. 8. Path diagram SmartPLS.

In this study, structural equation modeling was used to illustrate and evaluate the relationships among the variables in the proposed research model. The analysis distinguishes between three types of effects: direct effects, which represent the immediate influence of one variable on another; indirect effects, which capture the mediated influence through one or more intermediary variables; and total effects, which represent the combined magnitude of both direct and indirect effects.

TABLE III. STANDARD DIRECT EFFECTS

	US	CI
SQ	0.196	-0.040
EQ	0.193	0.111
IQ	-0.006	0.008
PA	0.197	0.237
EE	0.201	0.169
EC	0.157	-0.038
US		0.292

The results of the direct effect analysis are presented in Table III. Among the predictors of User Satisfaction (US), Effort Expectancy (EE) shows the largest direct effect with a standardized path coefficient of 0.201, followed by System Quality (SQ, 0.196) and Perceived Advantage (PA, 0.197). For Continuance Intention (CI), the variable with the strongest direct effect is User Satisfaction, with a coefficient of 0.292, confirming its mediating role between system-related constructs and behavioral intention.

TABLE IV. STANDARD INDIRECT EFFECTS

	CI
SQ	0.057
EQ	0.056
IQ	-0.002
PA	0.058
EE	0.059
EC	0.046

The results of the indirect effect analysis are shown in Table IV. The largest indirect effect on CI is contributed by Effort Expectancy (EE) with a coefficient of 0.059, closely followed by PA (0.058) and SQ (0.057). This indicates that these variables primarily influence CI through their effect on US, rather than directly.

TABLE V. STANDARD TOTAL EFFECTS

	US	CI
SQ	0.196	0.017
EQ	0.193	0.168
IQ	-0.006	0.007
PA	0.197	0.295
EE	0.201	0.228
EC	0.157	0.007
US		0.292

The total effects are summarized in Table V, which combines both direct and indirect effects. The findings reveal that PA has the strongest overall influence on CI (0.295), followed by EE (0.228) and EQ (0.168). These results highlight the importance of enhancing users' perceived benefits and ease of effort to improve both satisfaction and intention to continue using the system.

J. Hypothesis Test

The result of Structural Equation Modeling analysis using the SmartPLS V.4 application is shown in Table VI.

TABLE VI. HYPOTHESIS TEST

	O	T	P	Label
SQ→US	0,196	3,629	0	Significant
SQ→CI	-0,04	0,66	0,510	Not significant
SQ→US→CI	0,057	2,827	0,005	Significant
EQ→US	0,193	4,463	0	Significant
EQ→CI	0,111	1,768	0,077	Not significant
EQ→US→CI	0,056	3,191	0,001	Significant
IQ→US	-0,006	0,094	0,925	Not significant
IQ→CI	0,008	0,113	0,910	Not significant
IQ→US→ CI	-0,002	0,092	0,927	Not significant
PA→US	0,197	3,357	0,001	Significant
PA→CI	0,237	3,834	0	Significant
PA→US→CI	0,058	2,488	0,013	Significant
EE→US	0,201	3,291	0,001	Significant
EE→CI	0,169	2,543	0,011	Significant
EE→US→CI	0,059	2,396	0,017	Significant
EC→US	0,157	2,468	0,014	Significant
EC→CI	-0,038	0,59	0,555	Not significant
EC→US→CI	0,046	2,187	0,029	Significant
US→CI	0,292	4,303	0	Significant

H1a. Based on Table VI, summarizing the structural model evaluation, the path from System Quality (SQ) to User Satisfaction (US) shows a t-value of 3.629 with a p-value < 0.05, indicating a significant positive effect ($\beta = 0.196$). This means that higher system quality reflected in faster response times, system flexibility, and high availability contributes significantly to increased satisfaction among digital banking users. These findings highlight that technical reliability and smooth system performance are essential for improving positive user experience.

H1b. Based on Table VI, the path from System Quality (SQ) to Continuance Intention (CI) yields a t-value of 0.660 with a p-value of 0.510 (> 0.05), indicating no significant effect ($\beta = -0.040$). This suggests that while system quality influences satisfaction, it does not directly drive users' intention to continue using the mobile banking app. Instead, its influence may be more indirect, mediated by satisfaction.

H1c. Based on Table VI, the indirect effect of System Quality (SQ) on Continuance Intention (CI) through User Satisfaction (US) is significant ($t = 2.827$, $p = 0.005$, $\beta = 0.057$). This indicates that system quality impacts continuance intention indirectly by first enhancing user satisfaction, which then increases the likelihood of continued usage.

H2a. Based on Table VI, the path from Service Quality (EQ) to User Satisfaction (US) shows a t-value of 4.463 with a p-value < 0.05, indicating a significant positive effect ($\beta = 0.193$). This suggests that service attributes such as reliability, responsiveness, assurance, empathy, and tangibility play an important role in building user satisfaction with the mobile banking app.

H2b. Based on Table VI, the path from Service Quality (EQ) to Continuance Intention (CI) yields a t-value of 1.768 with a p-value of 0.077 (> 0.05), showing no direct significant effect ($\beta = 0.111$). This means that high-quality service alone may not directly encourage continued use unless it first enhances satisfaction.

H2c. Based on Table VI, the indirect effect of Service Quality (EQ) on Continuance Intention (CI) via User Satisfaction (US) is significant ($t = 3.191$, $p = 0.001$, $\beta = 0.056$). This shows that service quality drives continuance intention indirectly through its effect on satisfaction.

H3a. Based on Table VI, the path from Information Quality (IQ) to User Satisfaction (US) is not significant ($t = 0.094$, $p = 0.925$, $\beta = -0.006$), suggesting that timeliness, relevance, completeness, understandability, and consistency of information do not strongly influence satisfaction in this context.

H3b. Based on Table VI, the path from Information Quality (IQ) to Continuance Intention (CI) is also not significant ($t = 0.113$, $p = 0.910$, $\beta = 0.008$), indicating minimal influence of information quality on continued use of the app.

H3c. Based on Table VI, the indirect effect of Information Quality (IQ) on Continuance Intention (CI) via User Satisfaction (US) is not significant ($t = 0.092$, $p = 0.927$, $\beta = -0.002$), confirming that this variable does not play a substantial role in influencing continuance intention.

H4a. Based on Table VI, the path from Perceived Advantage (PA) to User Satisfaction (US) is significant ($t = 3.357$, $p = 0.001$, $\beta = 0.197$). This suggests that promotional benefits, attractive functionalities, and visual appeal contribute meaningfully to higher satisfaction levels.

H4b. Based on Table VI, the path from Perceived Advantage (PA) to Continuance Intention (CI) is significant ($t = 3.834$, $p < 0.05$, $\beta = 0.237$), indicating that perceived advantages directly encourage continued use of the digital banking app.

H4c. Based on Table VI, the indirect effect of Perceived Advantage (PA) on Continuance Intention (CI) via User Satisfaction (US) is also significant ($t = 2.488$, $p = 0.013$, $\beta = 0.058$), meaning that perceived advantages affect continuance intention both directly and indirectly.

H5a. Based on Table VI, the path from Effort Expectancy (EE) to User Satisfaction (US) is significant ($t = 3.291$, $p = 0.001$, $\beta = 0.201$), showing that ease of use, ease of learning, feature clarity, and language clarity play an important role in satisfaction.

H5b. Based on Table VI, the path from Effort Expectancy (EE) to Continuance Intention (CI) is significant ($t = 2.543$, $p = 0.011$, $\beta = 0.169$), suggesting that users who find the app easy to use are more likely to keep using it.

H5c. Based on Table VI, the indirect effect of Effort Expectancy (EE) on Continuance Intention (CI) via User Satisfaction (US) is significant ($t = 2.396$, $p = 0.017$, $\beta = 0.059$), reinforcing the role of satisfaction as a mediator.

H6a. Based on Table VI, the path from Digital Ecosystem (EC) to User Satisfaction (US) is significant ($t = 2.468$, $p = 0.014$, $\beta = 0.157$), indicating that lifestyle integration, multi-service access, and partner benefits positively influence satisfaction.

H6b. Based on Table VI, the path from Digital Ecosystem (EC) to Continuance Intention (CI) is not significant ($t = 0.590$, $p = 0.555$, $\beta = -0.038$), suggesting no direct influence on continuance intention.

H6c. Based on Table VI, the indirect effect of Digital Ecosystem (EC) on Continuance Intention (CI) via User Satisfaction (US) is significant ($t = 2.187$, $p = 0.029$, $\beta = 0.046$), indicating that ecosystem features still contribute indirectly by first improving satisfaction.

H7. Based on Table VI, the path from User Satisfaction (US) to Continuance Intention (CI) is significant ($t = 4.303$, $p < 0.05$, $\beta = 0.292$), confirming that satisfaction is a strong predictor of continued use of the mobile banking app.

V. Result and Discussion

A. Result

The analysis of the structural model reveals that five variables—System Quality (SQ), Service Quality (EQ), Perceived Advantage (PA), Effort Expectancy (EE), and Digital Ecosystem (EC)—exert significant positive effects on User Satisfaction (US), with p-values < 0.05. Among them, Effort Expectancy demonstrates the strongest effect ($\beta \approx 0.201$), highlighting that ease of use and clarity of features are the most critical determinants of satisfaction. Perceived Advantage ($\beta \approx 0.197$) follows closely, indicating that users place high value on promotions, special features, and cost-saving benefits offered by digital banks. System Quality ($\beta \approx 0.196$) and Service Quality ($\beta \approx 0.193$) also play significant roles, showing that technical reliability and responsive support are key enablers of a satisfying experience.

However, Information Quality (IQ) does not show a statistically significant effect on User Satisfaction ($p > 0.05$), suggesting that accuracy and completeness of information alone do not guarantee a better experience unless coupled with usability and relevance. When assessing Continuance Intention (CI), User Satisfaction emerges as the strongest direct predictor ($\beta \approx 0.292$, $p < 0.05$). Direct effects from SQ, EQ, IQ, and EC to CI are not significant, which indicates that these variables influence continued usage only through their contribution to satisfaction. Mediation analysis confirms that SQ, EQ, PA, EE, and EC all have significant indirect effects on CI through US, whereas IQ has neither direct nor meaningful mediated effect.

B. Discussion

These findings reinforce that User Satisfaction is the central mechanism driving continued use of digital banking applications. The dominant role of Effort Expectancy highlights the importance of designing seamless, intuitive interfaces—clear navigation, simple registration flows, and low-effort transactions strongly influence satisfaction.

Perceived Advantage is the next critical driver, underscoring that promotional benefits, competitive interest rates, and feature-rich offerings meaningfully improve user perceptions and satisfaction. This confirms that customers are motivated not just by functionality but by tangible value received.

System Quality and Service Quality contribute significantly as well. Reliable system performance, low downtime, and fast response times foster confidence, while responsive, helpful customer support ensures users feel supported. Service personalization and 24/7 assistance can further strengthen satisfaction levels.

Although Information Quality did not significantly impact satisfaction, this finding suggests that users may perceive information as a basic expectation rather than a differentiator. Digital banks should focus on contextualizing and personalizing information turning raw data into actionable insights that enhance decision-making to make it more influential.

Finally, the Digital Ecosystem plays a supportive but meaningful role, with its positive effect showing that integration with lifestyle services (QRIS payments, bill payments, e-commerce, insurance, investments) enhances satisfaction. Strengthening this ecosystem may deepen user engagement, especially when combined with superior usability and service quality.

Collectively, these results demonstrate that customer satisfaction is the linchpin variable, mediating most effects on continuance intention. This implies that improvements in technical, service, and functional dimensions will only translate into loyalty if they first enhance user satisfaction.

VI. Conclusion and Recommendation

A. Conclusion

This study result shows that there are significant or insignificant factors that affect with the Indonesian digital bank application. Based on the result of the analysis and hypothesis that has been done by researchers, the following conclusion is drawn,

1) The results show that System Quality has a significant effect on the User Satisfaction. By accpeting this hypothesis, it can be intepred that a responsive, reliable, and user-oriented system can increase customer satisfaction.

2) The results show that System Quality has no significant effect on Continuance Intention. This indicates that System Quality alone does not directly influence customers' intention to keep using the application.

3) The results show that System Quality, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that better system quality improves satisfaction, which in turn increases customers' intention to continue using the application.

4) The results show that Service Quality has a significant effect on User Satisfaction. This proves that fast, reliable, and empathetic customer service can improve customer satisfaction in digital bank.

5) The results show that Service Quality has no significant effect on Continuance Intention. This indicates that high service quality does not directly guarantee customers' intention to keep using the application.

6) The results show that Service Quality, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that high service quality increases satisfaction, which then strengthens the intention to continue using the app.

7) The results show that Information Quality has a significant effect on User Satisfaction. This proves that accurate, clear, and relevant information can increase customer satisfaction.

8) The results show that Information Quality has no significant effect on Continuance Intention. This indicates that information quality alone does not directly influence customers' intention to keep using the application.

9) The results show that Information Quality, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that good information quality improves satisfaction, which in turn increases continued usage intention.

10) The results show that Perceived Advantage has a significant effect on User Satisfaction. This proves that integrated features, useful benefits, and practical functionality can increase satisfaction in digital banking apps.

11) The results show that Perceived Advantage has a significant effect on Continuance Intention. This proves that tangible benefits and convenience encourage customers to keep using the application.

12) The results show that Perceived Advantage, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that perceived benefits raise satisfaction, which then increases continued usage intention.

13) The results show that Effort Expectancy has a significant effect on User Satisfaction. This proves that ease of use, clear features, and simple navigation can improve customer satisfaction.

14) The results show that Effort Expectancy has a significant effect on Continuance Intention. This proves that a simple and user-friendly application design motivates customers to keep using the application.

15) The results show that Effort Expectancy, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that ease of use raises satisfaction, which then influences continued usage.

16) The results show that Digital Ecosystem has a significant effect on User Satisfaction. This proves that integration with lifestyle services and partner platforms increases customer satisfaction.

17) The results show that Digital Ecosystem has a significant effect on Continuance Intention. This proves that a strong and integrated digital ecosystem directly motivates customers to keep using the application.

18) The results show that Digital Ecosystem, mediated by User Satisfaction, has a significant effect on Continuance Intention. This proves that a strong and integrated digital ecosystem boosts satisfaction, leading to higher user retention.

19) The results show that User Satisfaction has a significant effect on Continuance Intention. This proves that overall customer satisfaction directly influences the intention to keep using the digital banking application.

B. Recommendation

Based on the results of this research, Indonesian digital bank mobile applications can be further improved to increase customer satisfaction, which in turn will encourage customers' continuance intention. Improvement efforts should be prioritized according to their level of impact on user satisfaction, starting with Service Quality as the most influential factor, followed by Effort Expectancy, Perceived Advantage, System Quality, and finally Digital Ecosystem. This prioritization ensures that resources are focused on the areas with the highest potential to strengthen satisfaction and loyalty. Some suggestions are as follows:

1) Since customer satisfaction has a very high significant effect on continuance intention and serves as a key mediator, focusing on improving all variables that influence customer satisfaction will help ensure customers remain loyal to the company. In addition, efforts should be made to enhance information quality so that it can also contribute positively to customer satisfaction.

2) Given that Service Quality demonstrated the strongest influence on User Satisfaction, digital banks should prioritize improving responsiveness, reliability, and personalization of their services. This can be achieved through the provision of 24/7 customer support.

3) Given that Effort Expectancy have the second strongest influence on User Satisfaction, digital banks also need to improve their user interface by streamline application navigation, simplify account registration processes, and introduce features that minimize cognitive effort, such as biometric authentication, one-click transactions, and intuitive menu structures.

4) To strengthen perceived advantage, banks should actively communicate and expand unique benefits such as high-yield savings products, cashback and referral campaigns, and free transfers. They should also introduce advanced financial management tools such as budgeting dashboards and spending analysis, turning the app into a comprehensive financial management platform that keeps users engaged and loyal.

5) Although digital ecosystem had the smallest direct effect, it still contributes to user satisfaction and overall experience. Expanding integration with external services such as e-commerce platforms, insurance providers, and investment products can turn the app into a "one-stop" financial hub, increasing user engagement and reducing the need for multiple apps.

Future suggestions for subsequent research are to include aspects of information technology indicators or questions, as well as additional factors related to security. The study could also be continued using an updated research model with other variables that are more relevant to the digital banking services implemented by Indonesian digital banking applications.

Data Availability Statement

All the data of this research are available here (https://data.mendeley.com/datasets/6s94mxdt5t/2). The link contains the respondent's answer data. The illustrative result of the study analysis is fully disclosed in the tables and figures in this research.

Author's Contribution Statement

In this research, Soros Lie contributed to conceptualization, methodology, formal analysis, investigation, and writing. Viany Utami Tjhin contributed to supervision, project administration, funding acquisition, and writing review and editing. All authors have read and approved the submitted version of the manuscript and agree to be accountable for all aspects of the work.

References

[1] Kompas.com, "Digital crossover: Indonesia masuki era perbankan dalam genggaman di 2023," Kompas.com. Accessed: May 16, 2025. [Online]. Available: https://money.kompas.com/read/2022/12/23/063448726/digital-crossover-indonesia-masuki-era-perbankan-dalam-genggaman-di-2023?page=all

[2] E. R. Indriyarti, M. Christian, H. Yulita, T. Aryati, and R. J. Arsjah, "Digital Bank Channel Distribution: Predictors of Usage Attitudes in Jakarta's Gen Z," Journal of Distribution Science, vol. 21, no. 2, pp. 21–34, 2023, doi: 10.15722/jds.21.02.202302.21.

[3] K. K. R. Direktorat Jenderal Perbendaharaan, "Digital Banking," Kementerian Keuangan Republik Indonesia. Accessed: May 16, 2025. [Online]. Available: https://djpb.kemenkeu.go.id/direktorat/pkn/id/odading/2919-digital-banking.html

[4] SeaBank Indonesia, "SeaBank," Apple App Store. Accessed: Sep. 08, 2024. [Online]. Available: https://apps.apple.com/id/app/seabank/id1525477806

[5] SeaBank Indonesia, "SeaBank," Google Play. Accessed: Sep. 08, 2024. [Online]. Available: https://play.google.com/store/apps/details?id=id.co.bankbkemobile.digitalbank&hl=id

[6] PT Bank Jago Tbk, "Bank Jago/Jago Syariah," Apple App Store. Accessed: Sep. 08, 2024. [Online]. Available: https://apps.apple.com/id/app/bank-jago-jago-syariah/id1539402234

[7] PT Bank Jago Tbk, "Bank Jago/Jago Syariah," Google Play Store. Accessed: Sep. 08, 2024. [Online]. Available: https://play.google.com/store/apps/details?id=com.jago.digitalBanking

[8] A. Geebren, A. Jabbar, and M. Luo, "Examining the role of consumer satisfaction within mobile eco-systems: Evidence from mobile banking services," Comput Human Behav, vol. 114, p. 106584, Jan. 2021, doi: 10.1016/j.chb.2020.106584.

[9] Y. Vincenzo and R. Jayadi, "IMPORTANT FACTORS THAT AFFECT CUSTOMER SATISFACTION WITH DIGITAL BANKS IN INDONESIA," J Theor Appl Inf Technol, vol. 28, no. 4, 2023, [Online]. Available: www.jatit.org

[10] L. Chen, J. Jia, and C. Wu, "Factors influencing the behavioral intention to use contactless financial services in the banking industry: An application and extension of UTAUT model," Front Psychol, vol. 14, 2023, doi: 10.3389/fpsyg.2023.1096709.

[11] Otoritas Jasa Keuangan, "Peraturan Otoritas Jasa Keuangan Nomor 12/POJK.03/2020 tentang Konsolidasi Bank Umum," 2020. Accessed: Aug. 16, 2025. [Online]. Available: https://www.ojk.go.id/id/kanal/perbankan/regulasi/Pages/Peraturan-OJK-Nomor-12-POJK.03-2020.aspx

[12] G. Gerard and T. Dela Rosa, "Journal of Economics, Finance and Accounting Studies Trust and Satisfaction in Digital Banking in the Philippines-Integrating Delone and Mclean Information System Success Model," 2024, doi: 10.32996/jefas.

[13] O. M. A. Alqudah, B. A. F. Jarah, A. R. Alshehadeh, Z. Al-Matarneh, M. Z. Soda, and H. A. Al-Khawaja, "Data processing related to the impact of performance expectation, effort expectation, and perceived usefulness on the use of electronic banking services for customers of Jordanian banks," International Journal of Data and Network Science, vol. 7, no. 2, pp. 657–666, Mar. 2023, doi: 10.5267/j.ijdns.2023.3.006.

[14] J. N. Jurnawan and T. Oktavia, "ANALYSIS OF FACTORS INFLUENCING CONTINUANCE INTENTION TOWARDS DIGITAL BANK APPLICATIONS," J Theor Appl Inf Technol, vol. 31, no. 20, 2024, [Online]. Available: www.jatit.org

[15] A. A. Rabaa'i and S. A. ALMaati, "Exploring the Determinants of Users' Continuance Intention to Use Mobile Banking Services in Kuwait : Extending the Expectation-Confirmation Model," Asia Pacific Journal of Information Systems, vol. 31, no. 2, pp. 141–184, Jun. 2021, doi: 10.14329/apjis.2021.31.2.141.

[16] M. A. Memon, T. Ramayah, J. H. Cheah, H. Ting, F. Chuah, and T. H. Cham, "PLS-SEM STATISTICAL PROGRAMS: A REVIEW," Journal of Applied Structural Equation Modeling, vol. 5, no. 1, pp. i–xiv, 2021, doi: 10.47263/JASEM.5(1)06.

[17] Y. Ade, P. Siregar, and R. Jayadi, "CUSTOMER SATISFACTION TOWARD DIGITAL BANK SERVICES IN INDONESIA," The Seybold report, vol. 18, No 12, pp. 1789–1803, Dec. 2023, doi: 10.5281/zenodo.10444303.

[18] T. T. Nguyen, H. T. Nguyen, H. T. Mai, and T. T. M. Tran, "Determinants of digital banking services in Vietnam: Applying utaut2 model," Asian Economic and Financial Review, vol. 10, no. 6, pp. 680–697, Jun. 2020, doi: 10.18488/journal.aefr.2020.106.680.697.

[19] Infobanknews, "Seabank raup laba Rp204 miliar, melonjak 350 persen pada semester I 2024," Infobanknews. Accessed: Sep. 09, 2025. [Online]. Available: https://infobanknews.com/seabank-raup-laba-rp204-miliar-melonjak-350-persen-pada-semester-i-2024/

[20] Validnews, "Total nasabah Bank Jago capai 12 juta," Validnews. Accessed: Sep. 09, 2024. [Online]. Available: https://validnews.id/ekonomi/total-nasabah-bank-jago-capai-juta

[21] F. Nyimbili and L. Nyimbili, "Types of Purposive Sampling Techniques with Their Examples and Application in Qualitative Research Studies," British Journal of Multidisciplinary and Advanced Studies, vol. 5, no. 1, pp. 90–99, Feb. 2024, doi: 10.37745/bjmas.2022.0419.

[22] H. Sujati, Sajidan, M. Akhyar, and Gunarhadi, "Testing the Construct Validity and Reliability of Curiosity Scale Using Confirmatory Factor Analysis," Journal of Educational and Social Research, vol. 10, no. 4, p. 229, Jul. 2020, doi: 10.36941/jesr-2020-0080.

APPENDIX A
QUESTIONNAIRE ITEMS

Table VII lists the variables included in the research model along with their corresponding codes and indicators. This table provides a clear mapping between each variable and the specific measurement items used in the questionnaire, ensuring transparency and reproducibility of the research instrument. The complete wording of these measurement items, along with demographic and usage-experience questions, is presented in Table VIII.

TABLE VII. CONSTRUCT AND MEASUREMENT ITEMS

Variable	Code	Indicator
System Quality	SQ1	Response Time
	SQ2	Flexibility
	SQ3	Availability
Service Quality	EQ1	Tangibility
	EQ2	Reliability
	EQ3	Responsiveness
	EQ4	Assurance
	EQ5	Empathy
Information Quality	IQ1	Timeliness
	IQ2	Relevance
	IQ3	Completeness
	IQ4	Understandability
	IQ5	Consistency
Perceived Advantage	PA1	Promotional Benefit
	PA2	Functionalities
	PA3	Visual Appeal
Effort Expectancy	EE1	Ease of Use
	EE2	Ease of Learning
	EE3	Feature Clarity
	EE4	Language Clarity
Digital Ecosystem	EC1	Lifestyle Integration
	EC2	Multiservice Access
	EC3	Multi-Partner Access
	EC4	Partner Benefit Access
User Satisfaction	US1	System Satisfaction

	US2	Information Satisfaction
	US3	Recommendation Intention
Continuance Intention	CI1	Continuance Intention
	CI2	Actual Usage
	CI3	Preference Over Alternatives

Table VIII presents the full set of survey questions administered to respondents. Construct measurement items were rated on a 7-point Likert scale (1 = Strongly Disagree, 2 = Disagree, 3 = Somewhat Disagree, 4 = Neutral, 5 = Somewhat Agree, 6 = Agree, 7 = Strongly Agree) to capture the degree of agreement with each statement. Demographic and usage-experience questions were included for descriptive analysis only and were not part of the structural model.

TABLE VIII. SURVEY QUESTIONS

Section	Survey Question	Response Options / Scale
Profile / Demographic Questions	Gender	Male, Female
	Age	17 – 27 years, 28 – 43 years, 44 – 58 years, ≥ 59 years
	Occupation	Student, Private Employee, Civil Servant (PNS/ASN), Freelancer/Independent Worker, Entrepreneur, Homemaker, Informal Sector Worker (e.g., online driver, small trader, daily laborer), Retired
	Most Frequently Used Application	Bank Jago, SeaBank
Digital Bank Application Usage Experience	Duration of Use	Less than 1 month, 1–6 months, 7–12 months, 1–2 years, More than 2 years
	Frequency of Use	More than once a day, Once a day, Every 2–3 days, Every 4–6 days, Once a week, More than a week but less than two weeks, Every two weeks, Once a month
	Main Reason for Use	Comprehensive features, Promotional benefits (interest rate promos, free transfers, cashback), Easy account opening process, Application security & data privacy protection, No monthly admin fees, Attractive app design, Investment & financial management features, User-friendly interface, Clear & comprehensive information, Fast access and rarely encounters errors
Construct Measurement Items	SQ1 – The waiting time between pages or features in this application is very short	7-point Likert scale
	SQ2 – This application is easily adaptable to my financial needs	7-point Likert scale
	SQ3 – This application is easily accessible whenever I need it	7-point Likert scale
	EQ1 – The online customer service feature in this application is easy to use	7-point Likert scale
	EQ2 – Important features such as transfers and payments always work properly when I use them	7-point Likert scale
	EQ3 – The online customer service in this application provides quick assistance when I encounter problems	7-point Likert scale
	EQ4 – This application ensures the security of my banking data and transactions	7-point Likert scale
	EQ5 – I feel heard when I contact the online customer service in this application	7-point Likert scale

	IQ1 – Transaction and promotion information in this application is available promptly when needed	7-point Likert scale
	IQ2 – Information about promotions, transactions, and accounts provided by this application meets my needs	7-point Likert scale
	IQ3 – This application provides all information about my account and transactions completely	7-point Likert scale
	IQ4 – The information displayed by this application is clear and easy to understand	7-point Likert scale
	IQ5 – The information in this application is consistent across all sections (e.g., the amount, recipient name, or transaction time in the transaction list matches the details page)	7-point Likert scale
	PA1 – This application offers attractive promotions for me (e.g., free transfers, high interest rates, free e-wallet top-up, referral bonuses, no admin fees)	7-point Likert scale
	PA2 – This application has special features that help my financial activities (e.g., deposits, credit, auto-budgeting, spending analysis, joint accounts)	7-point Likert scale
	PA3 – The visual design of this application is appealing to me	7-point Likert scale
	EE1 – All features of this application are easy to use	7-point Likert scale
	EE2 – I found all features of this application easy to learn the first time I used it	7-point Likert scale
	EE3 – All features of this application are easy to understand	7-point Likert scale
	EE4 – The language used in this application is easy to understand	7-point Likert scale
	EC1 – I can access various services through this application (e.g., QRIS payments, electricity bills, internet packages, investment products, credit card, zakat, etc.)	7-point Likert scale
	EC2 – I can complete many needs — from paying bills to investing — directly in this application without switching to other apps	7-point Likert scale
	EC3 – This application is connected with many partners such as e-commerce, investment services, or insurance providers	7-point Likert scale
	EC4 – I often receive promos, discounts, or other benefits because this digital bank collaborates with many services or merchants	7-point Likert scale
	US1 – I am satisfied with the stability of this application's performance when I use it	7-point Likert scale
	US2 – I am satisfied with the quality of information provided by this application	7-point Likert scale
	US3 – I am willing to recommend this application to others	7-point Likert scale
	CI1 – I intend to continue using this application regularly	7-point Likert scale
	CI2 – I actively use this application in my daily activities	7-point Likert scale
	CI3 – I will continue to choose this application even if there are other alternatives available	7-point Likert scale

Generating a Trading Strategy Using Candlestick Patterns with Machine Learning

Hussaina Bala Malami[1], Badamasi Imam Ya'u[2]*, Fatima Umar Zambuk[3], Mohannad Alkanan[4], Osman Elwasila[5], Mohammad Shuaib Mir[6], Mohammed Nasir Danmalam Bawa[7], Yonis Gulzar[8]*

Department of Computer Science-Faculty of Computing, Abubakar Tafawa Balewa University Bauchi, Nigeria[1, 3]

Department of Computer Science-Faculty of Science, Islamic University in Uganda, Mbale, Uganda[2]

Department of Management Information Systems-College of Business Administration, King Faisal University, Al-Ahsa 31982, Saudi Arabia[4, 5, 6, 8]

Department of Computer Science, Niger State Polytechnic Zungeru, Nigeria[7]

***Abstract*—This study examines the application of machine learning (ML) algorithms for multi-day stock price prediction on the Nigerian Stock Exchange (NSE) from 2013 to 2023, to inform trading strategies. Utilizing candlestick patterns and technical indicators, including Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC), as input features, the models were trained to capture historical price dynamics. Among the evaluated algorithms, Ridge Regression demonstrated superior performance, achieving a Mean Absolute Error (MAE) of 0.0366 over a three-day forecasting horizon, while effectively mitigating overfitting and handling market volatility. In contrast, Decision Tree, Lasso, Support Vector Regressor (SVR), and K-Nearest Neighbors (KNN) models exhibited limitations due to sensitivity to data noise and overfitting. A recursive multi-step forecasting approach further enhanced prediction accuracy by incorporating temporal dependencies. However, backtesting revealed that predictive accuracy alone did not guarantee profitable trading outcomes, emphasizing the need to integrate market conditions, risk management, and strategy design. The findings underscore the importance of robust feature engineering and data preprocessing in financial ML applications. While Ridge Regression shows promise for stock price forecasting, successful trading strategies require a holistic framework that accounts for broader market factors. Future research should explore hybrid modeling techniques and additional exogenous variables to improve robustness.**

Keywords*—Nigerian stock trading; machine learning; pattern recognition; candlestick

I. Introduction

Artificial Intelligence (AI) is revolutionizing industries by solving complex problems and boosting efficiency. It powers advanced diagnostics and personalized care in healthcare [1-2], supports intelligent decision-making and resource optimization in agriculture [3–7], and creates safer, more efficient transportation systems through autonomous vehicles and traffic control [8]. In software development, AI automates coding tasks and enhances tools, accelerating innovation. Across the board, AI is a transformative force shaping the future of global industries [9-10].

Candlestick patterns originated in 18th-century Japan, where rice traders developed them as visual tools to represent price fluctuations [11]. The foundational principles underlying these patterns, particularly their ability to encode market sentiment and signal potential reversals, were established. Later, Western scholars such as Steve Nison systematized and popularized candlestick analysis globally, bridging Eastern and Western financial traditions [12], [13], [14], [15].

Stock market prediction remains a formidable challenge due to market noise and the semi-strong form of market efficiency, which posits that asset prices reflect all publicly available information [16]. Despite skepticism from financial economists about the feasibility of consistent profits through prediction, technical analysis, especially candlestick charting, has endured as a widely used methodology [17], [18]. Candlestick charts synthesize open, high, low, and close (OHLC) prices into intuitive visualizations, capturing both supply-demand dynamics [19] and investor psychology [20], [21]. Empirical studies have demonstrated their efficacy, for instance, [15] showed that candlestick-based strategies generated statistically significant profits in S&P 500 stocks, while [22] confirmed the profitability of one-day reversal patterns [23]. Subsequent work by [24] and [25] further validated the predictive power of two-day candlestick formations, and [26], [27] extended this to three-day patterns under varying market trends [28].

However, the literature is not unanimous. Contradictory findings, such as those of [29], who found no evidence of predictive ability in DAX index futures, underscore the need for further investigation [30]. These discrepancies motivate the integration of modern computational techniques, particularly machine learning (ML), to enhance candlestick analysis [31], [32]. ML's capacity to process high-dimensional data and detect non-linear patterns makes it well-suited to refine traditional methods.

II. Related Works

The research community has already investigated combining machine learning and pattern recognition to solve next-day stock price prediction in order to "take the best from the two worlds". For instance, Kamo and Dagli suggested a fuzzy logic-based gating network that accepts information regarding candlestick patterns as input [33]. To train feed-forward neural networks and support vector machines, respectively, approaches that extract features from candlestick charts were introduced [34][16]. An

*Corresponding Author, emails: biyau@iuiu.ac.ug, ygulzar@kfu.edu.sa

auto-regressive time series forecasting model with ordered fuzzy candlestick patterns was integrated [35].

On the other hand, researchers proposed PRML, a novel candlestick pattern recognition model using machine learning methods to improve stock trading decisions [11]. Four popular machine learning methods and 11 different feature types are applied to all possible combinations of daily patterns to start the pattern recognition schedule. Different time windows from one to ten days are used to detect the prediction effect at different periods. An investment strategy is constructed according to the identified candlestick patterns and a suitable time window.

A method that decouples the machine learning and pattern recognition steps within the trading system was proposed [31]. The study leverages various machine learning models, including both shallow and deep supervised models, as well as autoregressive techniques. These models are used to generate trading recommendations based on historical stock price data and relevant features.

Two authors proposed a technique to generate stock trading strategies employing features that are shown to effectively recognize patterns in candlestick charts [36]. The features are combined into a tree-like trading strategy using the Chi-square Automatic Interaction Detector algorithm. The technique is evaluated using actual stocks from the Stock Exchange of Thailand. The results show that the generated strategies are more profitable than other popular trading techniques, such as moving average convergence divergence, exponential moving average, relative strength index, stochastic oscillator, and average directional index.

Another researcher examined whether candlesticks patterns can predict trend swings and found that well-known 2-day "Engulfing" pattern has failed to produce a positive gain, while the "Harami" pattern has barely succeeded in doing so [37]. A more complex pattern known as the "Kicker" barely achieved a positive average gain and was also outperformed by the simple Buy and Hold (B&H) strategy. It was also reported that the "Stairs" pattern proposed achieved a positive gain for all twenty examined stocks and outperformed the B&H strategy for sixteen out of the twenty stocks.

A forecasting model based on chaotic mapping, firefly algorithm, and support vector regression to predict stock market price was proposed [16]. Compared with genetic algorithm-based SVR, chaotic genetic algorithm-based SVR, artificial neural networks, and adaptive neuro-fuzzy inference systems, the proposed model performs best on mean squared error and mean absolute percent error.

In another study, a stock price predicting system by combining SVR and an ensemble adaptive neuro fuzzy inference system (ENANFIS) was proposed [38]. The experimental results showed that the SVR-ENANFIS model has superior prediction performance than ENANFIS, SVR-Linear, SVR-SVR, and SVR-ANN.

III. Materials and Methods

A. Dataset Overview

Historical OHLC (Open, High, Low, Close) data from the Nigerian Stock Exchange (2013–2023) was collected. Using TA-Lib and pandas_ta libraries, 43 candlestick patterns were extracted. Technical indicators, SMA, EMA, and VROC, were also computed. Data preprocessing included handling missing values, linear interpolation, date indexing, and feature standardization.

A total of 51 features were created from the raw data. This included both traditional indicators and patterns (e.g. Doji, Hanging Man, Hammer). These features were selected to capture price direction, market momentum, and reversal likelihood.

Five ML algorithms were evaluated, which include Ridge Regression (with regularization), Decision Tree, Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN). The models were trained using Skforecast and Scikit-learn, optimized with time series cross-validation. Recursive multi-step forecasting was used for predicting three-day future prices. Our performance was assessed using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2). Furthermore, the hyperparameters for Ridge (alpha) and Decision Tree (depth, splits, leaves) were tuned using grid search.

B. Evaluation Metrics

To assess the accuracy and reliability of the machine learning models, three standard regression evaluation metrics were employed: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). Each provides unique insights into the model's predictive performance.

1) Root Mean Square Error (RMSE): RMSE measures the average magnitude of the errors between predicted and actual values, giving more weight to large errors due to the squaring of differences. A lower RMSE indicates better predictive accuracy. Because it penalizes large errors more heavily, RMSE is particularly useful when large deviations are especially undesirable in applications like stock price forecasting. RMSE is given by the following formula:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (1)$$

a) Mean Absolute Error (MAE): MAE represents the average absolute difference between the predicted and actual values, treating all errors equally regardless of direction. MAE is straightforward to interpret. A lower MAE reflects higher accuracy, making it useful for understanding the average error size in the same units as the target variable (e.g., price in Naira). MAE is given by the following formula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\lceil y_i - \hat{y}_i \rceil \quad (2)$$

b) Coefficient of determination (R^2): R^2 measures how well the model explains the variance in the actual data. It ranges from 0 to 1, with higher values indicating a better fit. An R^2 value close to 1 indicates that the model explains most of the variability in the target variable. In stock forecasting, a high R^2 suggests that the model captures significant trends and fluctuations in price data. R^2 is given by the following formula:

$$R^2 = 1 - \frac{\Sigma(y_i - \hat{y}_i)^2}{(y_i - \bar{y}_i)^2} \quad (3)$$

C. Proposed Model

The methodology begins with the collection of historical data from the Nigerian Stock Exchange, covering a 10-year period from 2013 to 2023. The dataset includes standard OHLC (Open, High, Low, Close) and volume data, which serve as the foundation for all subsequent analysis. Using libraries such as TA-Lib and pandas_ta, various technical indicators and candlestick patterns were extracted to enrich the dataset. In the feature engineering stage, 43 candlestick patterns (e.g. Doji, Hammer, Hanging Man) were identified, alongside key technical indicators like the Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC). Together, these contributed to a total of 51 features, carefully designed to capture short-term price movements, trend reversals, and market momentum.

Next, the data underwent preprocessing to ensure accuracy and compatibility with machine learning models. This included handling missing values through imputation, applying linear interpolation, setting proper date indexing, and standardizing the features to bring all values into a similar range. These steps were essential to improve model learning and prevent bias caused by scale differences. The model training and selection phase involved testing five machine learning algorithms: Ridge Regression, Lasso Regression, Decision Tree, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN). The training process utilized Scikit-learn and Skforecast libraries, with model hyperparameters optimized through grid search, particularly focusing on tuning aspects like Ridge's alpha and Decision Tree's depth and split criteria.

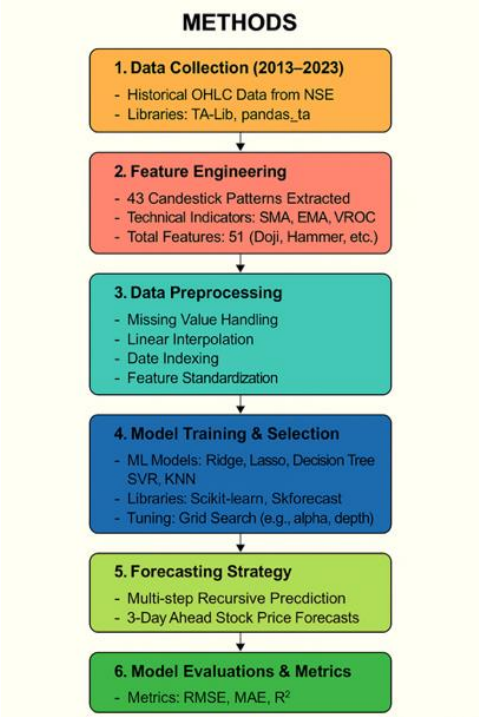


Fig. 1. The proposed model.

After training, the models were used in a multi-step recursive forecasting strategy to predict stock prices three days ahead. This approach allowed the system to simulate real trading conditions by using previous predictions as inputs for subsequent forecasts, which is particularly useful in time-series scenarios.

Finally, the models were subjected to evaluation using multiple performance metrics, including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). These metrics enabled a comprehensive assessment of each model's predictive accuracy, generalization ability, and error magnitude, supporting the selection of the most reliable model for further trading strategy development. Fig. 1 describes the processes involved in the proposed model.

IV. Results and Discussion

A. Predictive Performance

Among the five machine learning models evaluated, Ridge Regression demonstrated the most consistent and accurate performance in forecasting multi-day stock price movements. With a Mean Absolute Error (MAE) of 0.0366 and a Root Mean Square Error (RMSE) of 0.0367, Ridge Regression successfully balanced bias and variance through regularization, making it well-suited for the high-dimensional, noisy data typical of stock markets. Its strong generalization capabilities make it a robust candidate for short-term forecasting in volatile environments like the Nigerian Stock Exchange (NSE).

In contrast, the Decision Tree model, despite showing near-perfect accuracy on the training dataset (very low training error), suffered from severe overfitting. The test phase MAE rose sharply to 19.43, indicating a lack of generalization and an inability to handle new, unseen data. This result highlights the Decision Tree's sensitivity to noise and its tendency to overfit in the absence of pruning or ensemble techniques.

Other models, including Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN), delivered moderate to poor results. These models showed higher error rates in both training and testing phases, possibly due to their sensitivity to market volatility and the relatively sparse or nonlinear nature of the input data. KNN, in particular, may have been impacted by local noise, while SVR's performance is often highly dependent on.

B. Model Training

Fig. 2 illustrates the model's performance using the Root Mean Square Error (RMSE). The Ridge model performed best, with an RMSE of 1.35, followed by the Decision Tree model, which had an RMSE of 8.71. The Lasso, SVR, and KNN models performed poorly, with RMSE values of 24.94, 62.91, and 55.04, respectively.

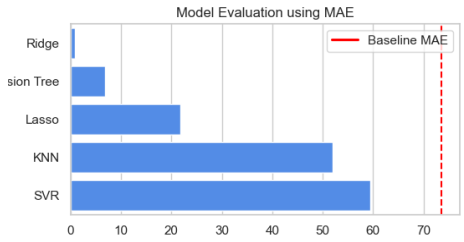


Fig. 2. Model evaluation using RMSE.

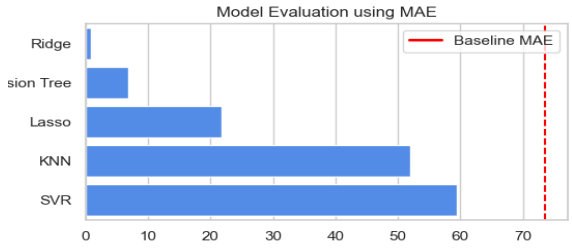


Fig. 3. Model evaluation using MAE.

Fig. 3 illustrates the model's performance using the Root Mean Square Error (RMSE). The Ridge model performed best, with an RMSE of 1.35, followed by the Decision Tree model, which had an RMSE of 8.71. The Lasso, SVR, and KNN models performed poorly, with RMSE values of 24.94, 62.91, and 55.04, respectively.

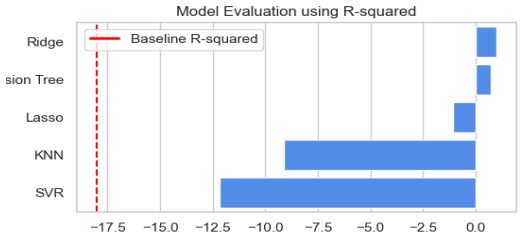


Fig. 4. Model evaluation using R-squared.

Fig. 4 shows how well the models performed using the coefficient of determination (R^2) metric. The Ridge model performed best with an R^2 of 0.99. The Decision Tree model also performed well with an R^2 of 0.75. However, the Lasso, SVR, and KNN models performed poorly, with R2 values of -1.07, -12.18, and -9.09, respectively, indicating no predictive power.

C. Trading System Implementation

Following the development of predictive models, a basic trading strategy was implemented to test the real-world applicability of the forecasts. The strategy used a fixed stop-loss of 3% and allocated 50% of available capital per trade, with signals generated from the Ridge model's predicted direction. The signal generator incorporated trend validation and position sizing logic based on forecasted returns.

However, early backtesting results indicated zero or negligible profit, despite strong predictive metrics. This disconnect between forecast accuracy and trading profitability highlights a critical gap in model-to-strategy translation. The lack of gains may be attributed to several factors: static risk parameters, market slippage, absence of transaction cost modeling, or timing mismatches between prediction and execution. These results underscore the importance of integrating dynamic risk management, execution optimization, and broader contextual signals (such as macroeconomic news or investor sentiment) into the trading framework.

D. Evaluation and Comparative Performance of Models

The comparative performance in Table I highlights how different machine learning models fared in predicting multi-day stock prices. Ridge Regression emerged as the most reliable model, achieving the lowest test MAE (0.0366) and RMSE (0.0367). Its regularization technique helped manage overfitting, making it well-suited for the noisy, multicollinear features typical in financial data. In contrast, the Decision Tree model, while showing excellent results during training, completely failed to generalize, with its test MAE skyrocketing to 19.43, indicating severe overfitting. The overfitting occurred because the decision tree was overly complex, memorizing noise and specific patterns in the training data rather than learning generalizable trends. In financial forecasting, this limits the model's usefulness, making regularization and robust validation essential to improve performance on unseen data.

Models like Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN) performed moderately. Lasso offered some interpretability and handled sparse features, but at a slight cost to accuracy. SVR showed reasonable balance, though sensitive to kernel tuning, while KNN struggled with market volatility due to its reliance on proximity in high-dimensional space. Overall, the results reinforce Ridge Regression as the most stable and generalizable model for short-term stock prediction in this context. However, they also underscore the importance of avoiding overly complex or overly simple models that may not translate well into real trading scenarios.

The poor performance of Lasso, SVR, and KNN likely stems from their sensitivity to the dataset's structural characteristics. Lasso's aggressive feature selection may have eliminated key predictors in a volatile, high-noise financial dataset, leading to severe underfitting. SVR and KNN, which rely heavily on local patterns and data density, struggle when market fluctuations create irregular, non-repetitive trends, making it difficult for them to capture meaningful predictive relationships.

TABLE I THE COMPARATIVE PERFORMANCE

Model	Training MAE	Testing MAE	Training RMSE	Testing RMSE	R^2 Score	Notes
Ridge Regression	0.0333	0.0366	0.0341	0.0367	High	Best overall performance; balanced generalization and low variance
Lasso Regression	0.0402	0.0601	0.0417	0.0632	Moderate	Penalizes overfitting but loses accuracy with sparse signals
K-Nearest Neighbors	0.0515	0.0560	0.0527	0.0583	Low	Struggles with volatility; sensitive to data density
SVR (Linear Kernel)	0.0498	0.0534	0.0505	0.0550	Moderate	Good for stable trends, less effective under erratic price movements
Decision Tree	**0.0005**	**19.43**	0.0006	19.88	Very Low	Severe overfitting; excellent on train data, but fails to generalize

V. CONCLUSION

This study investigated the application of machine learning algorithms in forecasting stock prices on the Nigerian Stock Exchange (NSE) using historical data from 2013 to 2023. The core objective was to develop a multi-day trading strategy by incorporating both candlestick patterns and technical indicators into predictive models. The integration of indicators such as the Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC), alongside 43 candlestick patterns, enriched the feature space and improved the models' ability to learn meaningful trends and price movements.

Among the five machine learning models evaluated, Ridge Regression consistently outperformed others, achieving a minimum Mean Absolute Error (MAE) of 0.0366 over a three-day forecast horizon. Its strong generalization capability and robustness to multicollinearity made it particularly well-suited for volatile financial datasets. In contrast, the Decision Tree model showed signs of overfitting, while Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN) were less effective due to their sensitivity to market fluctuations.

The study further emphasizes that accurate predictions alone do not guarantee profitable trading outcomes. While multi-step recursive forecasting proved effective in enhancing predictive precision, the transition from forecasts to profitable trades remains complex. Factors such as real-time market conditions, execution delays, and inadequate risk management can erode trading performance. This highlights the need for integrated strategies that combine robust predictive models with sound risk management, position sizing, and decision filters.

Ultimately, the research affirms the feasibility of leveraging machine learning, candlestick analysis, and technical indicators for short-term price prediction in emerging markets. It also underscores the importance of comprehensive data preprocessing, thoughtful feature engineering, and iterative model tuning. The findings contribute to the expanding literature on AI-driven financial analytics and offer a strong foundation for future exploration into adaptive trading systems.

A. Future Work

To further improve the effectiveness and practical relevance of the study, the following areas are recommended for future research and development:

1) Enhance Feature Engineering by adding more technical indicators (e.g. MACD, RSI) and refining candlestick representations to better capture market behavior.

2) Broaden the Dataset to include various sectors or global markets, improving the model's adaptability and generalizability.

3) Adopt Deep Learning Techniques such as LSTM or TCNs to better capture time-series patterns and enhance forecast accuracy.

4) Implement Dynamic Model Updating to ensure the system adapts to evolving market conditions and avoids performance degradation.

5) Expand Indicator Selection to identify and utilize the most predictive technical signals.

6) Optimize and Combine Models using ensemble or hybrid methods to improve robustness and predictive power.

7) Integrate Risk Management features like dynamic stop-loss, position sizing, and sentiment analysis to minimize trading risk.

8) Establish Continuous Evaluation through regular backtesting, real-time validation, and model recalibration to maintain performance over time.

ACKNOWLEDGMENT

This work was supported by the Deanship of Scientific Research, the Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia under the Project KFU253264.

REFERENCES

[1] N. M. Wormi, B. I. Ya'u, S. Boukari, M. A. Musa, F. Shittu, and M. A. Lawal, "Deeper Architecture for Brain Age Prediction Based on MRI Images Using Transfer Learning Technique," *Procedia Comput Sci*, vol. 212, pp. 441–453, 2022.

[2] M. A. Wakili, H. A. Shehu, M. Abdollahi, Y. B. Imam, M. H. Sharif, and H. Kusetogullari, "DenseNet-FPA: Integrating DenseNet and Flower Pollination Algorithm for Breast Cancer Histopathology Image Classification," *IEEE Access*, 2025.

[3] Gulzar, Y. PapNet: An AI-Driven Approach for Early Detection and Classification of Papaya Leaf Diseases. *Applied Fruit Science 2025 67:4* 2025, *67*, 1–11, doi:10.1007/S10341-025-01466-9.

[4] Gulzar, Y. Papaya Leaf Disease Classification Using Pre-Trained Deep Learning Models: A Comparative Study. *Applied Fruit Science* 2025, *67*, 1–10, doi:10.1007/S10341-025-01533-1/METRICS.

[5] Gulzar, Y.; Ünal, Z. Time-Sensitive Bruise Detection in Plums Using PlmNet with Transfer Learning. *Procedia Comput Sci* 2025, *257*, 127–132, doi:10.1016/J.PROCS.2025.03.019.

[6] Gulzar, Y.; Ünal, Z. Optimizing Pear Leaf Disease Detection Through PL-DenseNet. *Applied Fruit Science* 2025, *67*, 1–13, doi:10.1007/s10341-025-01265-2.

[7] Seelwal, P.; Dhiman, P.; Gulzar, Y.; Kaur, A.; Wadhwa, S.; Onn, C.W.; Goyal, N.; Shanmugasundaram, H. A Systematic Review of Deep Learning Applications for Rice Disease Diagnosis: Current Trends and Future Directions. *Front Comput Sci* 2024, *6*, 1452961, doi:10.3389/FCOMP.2024.1452961.

[8] Idris, I.Y.; Ya'u, B.I.; Ali, U.; Gulzar, Y. An Improved Deep Learning Framework Based on Multi-Scale Convolutional Architecture for Road Crack Detection. *Lecture Notes in Networks and Systems* 2025, *1165*, 109–121, doi:10.1007/978-981-97-8336-6_9.

[9] A. A. Aihong, B. I. Ya'U, U. Ali, A. Ahmad, and M. A. Lawal, "An optimized deep learning method for software defect prediction using Whale Optimization Algorithm," *Научно-технический вестник информационных технологий, механики и оптики*, vol. 24, no. 2, pp. 222–229, 2024.

[10] Z. Bala *et al.*, "Transfer learning approach for malware images classification on android devices using deep convolutional neural network," *Procedia Comput Sci*, vol. 212, pp. 429–440, 2022.

[11] Y. Lin, S. Liu, H. Yang, H. Wu, and B. Jiang, "Improving stock trading decisions based on pattern recognition using machine learning technology," *PLoS One*, vol. 16, no. 8, p. e0255558, 2021.

[12] Y. Hu, B. Feng, X. Zhang, E. W. T. Ngai, and M. Liu, "Stock trading rule discovery with an evolutionary trend following model," *Expert Syst Appl*, vol. 42, no. 1, pp. 212–222, 2015.

[13] W. Hu, Y.-W. Si, S. Fong, and R. Y. K. Lau, "A formal approach to candlestick pattern classification in financial time series," *Appl Soft Comput*, vol. 84, p. 105700, 2019.

[14] H. Hu, L. Tang, S. Zhang, and H. Wang, "Predicting the direction of stock markets using optimized neural networks with Google Trends," *Neurocomputing*, vol. 285, pp. 188–195, 2018.

[15] G. Caginalp and H. Laurent, "The predictive power of price patterns," *Appl Math Finance*, vol. 5, no. 3–4, pp. 181–205, 1998.

[16] E. Ahmadi, M. Jasemi, L. Monplaisir, M. A. Nabavi, A. Mahmoodi, and P. A. Jam, "New efficient hybrid candlestick technical analysis model for stock market timing on the basis of the Support Vector Machine and Heuristic Algorithms of Imperialist Competition and Genetic," *Expert Syst Appl*, vol. 94, pp. 21–31, 2018.

[17] R. C. Brasileiro, V. L. F. Souza, B. J. T. Fernandes, and A. L. I. Oliveira, "Automatic method for stock trading combining technical analysis and the artificial bee colony algorithm," in *2013 Ieee congress on evolutionary Computation*, IEEE, 2013, pp. 1810–1817.

[18] O. Bustos and A. Pomares-Quimbaya, "Stock market movement forecast: A systematic review," *Expert Syst Appl*, vol. 156, p. 113464, 2020.

[19] H. Bessembinder and K. Chan, "Market efficiency and the returns to technical analysis," *Financ Manage*, pp. 5–17, 1998.

[20] B. R. Marshall, M. R. Young, and L. C. Rose, "Candlestick technical trading strategies: Can they create value for investors?," *J Bank Financ*, vol. 30, no. 8, pp. 2303–2323, 2006.

[21] R. Naranjo, J. Arroyo, and M. Santos, "Fuzzy modeling of stock trading with fuzzy candlesticks," *Expert Syst Appl*, vol. 93, pp. 15–27, 2018.

[22] Y.-J. Goo, D.-H. Chen, and Y.-W. Chang, "The application of Japanese candlestick trading strategies in Taiwan," *Investment Management and Financial Innovations*, no. 4, Iss. 4, pp. 49–79, 2007.

[23] Q. Wang, W. Xu, and H. Zheng, "Combining the wisdom of crowds and technical analysis for financial market prediction using deep random subspace ensembles," *Neurocomputing*, vol. 299, pp. 51–61, 2018.

[24] M. Zhu, S. Atri, and E. Yegen, "Are candlestick trading strategies effective in certain stocks with distinct features?," *Pacific-Basin Finance Journal*, vol. 37, pp. 116–127, 2016.

[25] S. Chen, S. Bao, and Y. Zhou, "The predictive power of Japanese candlestick charting in Chinese stock market," *Physica A: Statistical Mechanics and its Applications*, vol. 457, pp. 148–165, 2016.

[26] T.-H. Lu, "The profitability of candlestick charting in the Taiwan stock market," *Pacific-Basin Finance Journal*, vol. 26, pp. 65–78, 2014.

[27] T.-H. Lu, Y.-C. Chen, and Y.-C. Hsu, "Trend definition or holding strategy: What determines the profitability of candlestick charting?," *J Bank Financ*, vol. 61, pp. 172–183, 2015.

[28] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer, "Deep learning for financial applications: A survey," *Appl Soft Comput*, vol. 93, p. 106384, 2020.

[29] J. H. Fock, C. Klein, and B. Zwergel, "Performance of candlestick analysis on intraday futures data," *Journal of Derivatives*, vol. 13, no. 1, p. 28, 2005.

[30] F. Zhou, Q. Zhang, D. Sornette, and L. Jiang, "Cascading logistic regression onto gradient boosted decision trees for forecasting and trading stock indices," *Appl Soft Comput*, vol. 84, p. 105747, 2019.

[31] L. Cagliero, J. Fior, and P. Garza, "Shortlisting machine learning-based stock trading recommendations using candlestick pattern recognition," *Expert Syst Appl*, vol. 216, p. 119493, 2023.

[32] L. Cagliero, P. Garza, G. Attanasio, and E. Baralis, "Training ensembles of faceted classification models for quantitative stock trading," *Computing*, vol. 102, no. 5, pp. 1213–1225, 2020.

[33] T. Kamo and C. Dagli, "Hybrid approach to the Japanese candlestick method for financial forecasting," *Expert Syst Appl*, vol. 36, no. 3, pp. 5023–5030, 2009.

[34] M. Jasemi, A. M. Kimiagari, and A. Memariani, "A modern neural network model to do stock market timing on the basis of the ancient investment technique of Japanese Candlestick," *Expert Syst Appl*, vol. 38, no. 4, pp. 3884–3890, 2011.

[35] A. Marszałek and T. Burczyński, "Modeling and forecasting financial time series with ordered fuzzy candlesticks," *Inf Sci (N Y)*, vol. 273, pp. 144–155, 2014.

[36] S. Thammakesorn and O. Sornil, "Generating trading strategies based on candlestick chart pattern characteristics," in *Journal of Physics: Conference Series*, IOP Publishing, 2019, p. 12008.

[37] G. Cohen, "Best candlesticks pattern to trade stocks," *International Journal of Economics and Financial Issues*, vol. 10, no. 2, p. 256, 2020.

[38] J. Zhang, L. Li, and W. Chen, "Predicting stock price using two-stage machine learning techniques," *Comput Econ*, vol. 57, no. 4, pp. 1237–1261, 2021.

Advancing Speech Enhancement with Generative Adversarial Network-Autoencoder: A Robust Adversarial Autoencoder Approach

Mandar Diwakar[1], Brijendra Gupta[2]
Department of Computer Engineering, Smt. Kashibai Navale College of Engineering, Pune, India[1]
Savitribai Phule Pune University, Pune, India[1]
Department of Information Technology, Siddhant College of Engineering, Pune, India[2]

***Abstract*—In day-to-day life, the speech signals are often noisy and distorted by background noise. These signals are not suitable for use in different audio-operated applications directly, as they are distorted. The use of these noisy voice signals can degrade the performance of the speech communication system. There are a huge number of applications nowadays that we use for various purposes, which utilize voice as input. Our study focuses on speech enhancement, which involves a combination of Generative Adversarial Networks (GAN) and Autoencoders (AE). The required features are extracted by using the MFCC algorithm from the MUSAN dataset. The features extracted with MFCC are paired samples of clean and noisy speech. The main architecture is a combination of GAN and AE. The Generator is trained to reconstruct clean speech features from noisy speech signal inputs. On the other hand, the discriminator is trained to tell the difference between real clean samples and samples that are generated by the generator. The adversarial training approach continuously improves the performance of the generator to produce good-quality and more intelligent speech. The MUSAN dataset used for the experiment contains data of noisy speech. As a result, we note that the model performs very well and shows robustness across multiple types of noise samples. The AE is used for feature reconstruction, and the GAN for generating fake samples. This combination of GAN and AE resulted in a good solution for speech enhancement in a noisy and distorted environment.**

Keywords*—Speech enhancement; Generative Adversarial Network (GAN); Autoencoder (AE); MFCC; noise robustness; adversarial training

I. Introduction

Speech communication has become common in modern society. There are multiple applications or features we are using in daily life that work on speech as input. For instance, take the example of Google search. When you tap on the microphone icon in Google search, it will allow you to fetch the speech input to search for certain queries. Whatever you fetch by using a speech signal will be converted into text, and the search result will be displayed on the screen. Similarly, you can see multiple applications such as Alexa by Amazon and Siri by Apple. These systems required smooth and clean speech input to run the systems and generate the output. These systems may struggle when exposed to noisy or distorted background noise, such as household appliances, outdoor sounds from streets, or any machinery working sound. Even though a low-intensity noise signal leads to a noticeable decline in clarity, smoothness, and originality of the speech signal transmitted [1].

Today, speech enhancement can be done with the help of newly emerged systems using GAN. This architecture consists of two neural network generators and a discriminator. The generator is responsible for generating, creating, or reconstructing the clean speech. At the same time, the discriminator is used to differentiate between real speech samples and those generated by the generator [2]. In this study, we integrate AE into a conventional GAN architecture for the improvement of speech enhancement. Performance of the model can be improved because AE has the capabilities of reconstruction of samples and generative fidelity of GANs. In this system, we are using MFCCs rather than waveforms to reduce computational complexities. In adversarial training, the generator is tasked to map noisy speech to clean features while the discriminator is trained to differentiate generated features that resemble real clean speech. This approach of the model improves robustness in low-SNR conditions. GAN-AE framework performs exceptionally well in speech enhancement by adding reconstruction with adversarial learning [3]. Our proposed approach integrates adversarial learning (GAN) with reconstruction learning (AE), ensuring both realism from the GAN and accurate signal reconstruction from the AE. The feature extraction was done with MFCC, and the extracted MFCC embeddings were used instead of raw waveforms. These embeddings represent a compact as well as meaningful representation of the speech features. The results show that the proposed approach exhibits excellent performance in low SNR and non-stationary noisy and distorted environments compared to the GAN variants. The Encoder and Decoder structure (AE) reduces computations, which is a main requirement for real-time speech enhancement.

The remainder of this paper is organized as follows. Section II gives information about the literature review on GAN-based speech enhancement systems. Section III describes the architecture of the proposed GAN-AE speech enhancement system. Section IV gives detailed information on the experimental setup used for model training. Section V describes the results generated by the experiment performed. And finally, the conclusion of this whole study is described in Section VI.

II. LITERATURE SURVEY

Pascual et al. SEGAN, i.e. speech enhancement Generative Adversarial Network, does not require MFCC features or special features to operate. SEGAN operates on raw audio waveforms directly from audio speech samples. In SEGAN, a generator is used as an AE that maps noisy speech to clean audio speech. Discriminator uses a CNN-based classifier to differentiate between real audio speech samples and generated samples. One of the limitations of SEGAN is that it requires specific GPU resources while training, and it also takes a long time to train. Some results also suggest that as SEGAN enhances raw audio speech, phase distortion is not handled effectively, which affects the clarity of speech samples [4]. Iterated Speech Enhancement Generative Adversarial Network (ISEGAN) and Deep Speech Enhancement Generative Adversarial Network (DSEGAN) are multi-generator GANs with an advanced SEGAN architecture used for improving speech enhancement. ISEGAN and DSEGAN use a unique technique to refine the noisy input by making a chain of multiple generators. ISEGAN iteratively applies the same enhancement mapping by using a shared parameter generator. DSEGAN enables diverse enhancement stages by employing independent generators with distinct parameters. ISEGAN takes high training time as it has a deeper architecture, and also increases GPU memory usage due to longer sequences [5]. TFDense-GAN is a unique framework where a time-frequency domain model uses a combo of a U-net-based Autoencoder and a multi-spectrogram discriminator. In this framework, robust noise reduction will be achieved by using a time-frequency transformer with a dense block. TFDense-GAN framework is considered one of the high-memory consumption models. This is due to the usage of dense block and transformer as a combination, which also causes slow inference. It's become quite challenging to deploy on edge devices due to their computational latency [6]. Arjovsky invented the Wasserstein Generative Adversarial Network (WGAN) as a variant architecture of the conventional GAN framework. The primary objective of WGAN is to enhance training stability and address issues such as mode collapse and vanishing gradients, which are common in GAN architectures. WGAN, as an improved variant of GAN, still requires careful training of hyperparameters like learning rate and weight clipping range [7]. CycleGAN (Cycle-Consistent Generative Adversarial Network) is a Generative Adversarial Network that works on unsupervised speech samples. CycleGAN is capable to learn a mapping between noisy speech samples and clean ones. There is no need for a parallel noisy & clean dataset. CycleGAN works with the principle of two generators and two discriminators, which can lead to producing residual noise in enhanced speech [8]. CinCGAN-SE (Cycle-in-Cycle GAN for Speech Enhancement) is a GAN framework designed to overcome two critical limitations of traditional speech enhancement systems. The first one is an unpaired data requirement, and the other is phase-aware enhancement. It's also useful in addressing issues regarding residual noise and phase distortion. In speech enhancement, CinCGAN may face phase reconstruction challenges in low SNR conditions [9].

DiscoGAN (Discovering Cross-Domain Relations with GANs) has been introduced to convert whispered speech to clean-voiced speech. DiscoGAN is a dual-generator GAN architecture that works on unsupervised samples. This architecture may face issues with mode collapse and computational cost [10][11]. The convolutional neural network (CNN) is widely known for its success in speech recognition and enhancement. CNN has proven effective speech recognition by processing voice signals as spectrograms or MFFCs. Raw speech is converted into waveforms or a digital signal and then processed [12]. CNN contains multiple layers, like convolution, pooling, and fully connected layers. Local speech patterns get detected by a convolution layer from spectrograms. It could be achieved by applying filters across the time-frequency domain. A pooling layer is used for down-sampling the features to reduce the computational load while maintaining the key acoustic information. The role of the fully connected layer is to classify extracted features into phonemes for the outcomes. There is one more important component of CNN, CTC, i.e. Connectionist Temporal Classification, used for solving the alignment between variable-length audio inputs and outputs by allowing flexible mapping during the training time [13]. Apart from the well-known success in speech enhancement, the CNN has some limitations. CNN may struggle when exposed to long-range dependencies in the speech signal. CNN may lead to miss variable-length speech patterns like prolonged vowels against short stops. This may happen as a convolution kernel has a fixed-size window. It is also observed that most of the speech systems that are based on CNN are highly dependent on preprocessed features rather than raw speech input [14]. A Deep Neural Network (DNN) is another framework from which speech enhancement is possible. Firstly, features are extracted from the noisy speech, such as MFCC, Log-mel spectrogram, and SIFT magnitude. Now these extracted features feed into the input layers of the DNN [15]. Each hidden layer learns a distinguished abstract pattern, such as speech phoneme shapes, noise patterns, and silence segments. DNN learns to transform noisy speech samples to clean ones by using some activation functions like ReLU. It learns how to map noisy input to clean output. The output of the DNN is denoised speech samples resembling clean features. Then, in inverse feature transformation, all features convert back to audio formats. At output, we get the final waveform with the denoised audio signal [16].

Table I shows a comparative table that gives summarized information on all GAN variants.

Looking at these variants, each addresses specific limitations, but none fully balances computational efficiency, robustness to diverse noise types, phase-aware enhancement, and ease of training. Moreover, existing methods often focus either on strong generative modeling or precise reconstruction, but not both in a balanced manner. The main goal of the GAN-AE model is to deliver robust, perceptually pleasing speech enhancement with reduced architecture overhead. This also addresses the trade-offs observed in previous GAN-based methods.

TABLE I COMPARATIVE ANALYSIS OF GAN ARCHITECTURES FOR SPEECH ENHANCEMENT: METHODS, ADVANCES, AND LIMITATIONS [17] [18] [19] [20] [21]

GAN Variant	Methodology	Advantages	Limitations	Improvements
SEGAN	Raw waveform U-Net + CNN discriminator Adversarial + L1 loss	Phase preservation End-to-end training	Struggles with high noise High computational cost	First GAN for raw waveform enhancement Introduced adversarial learning for speech
ISEGAN/ DSEGAN	Multi-generator refinement (Iterative/Deep cascaded)	Gradual noise suppression Better PESQ/STOI	Memory intensive Training complexity	Added iterative refinement Reduced artifacts through multi-stage processing
TFDense-GAN	Time-frequency U-Net + Dense Blocks multi-spectrogram discriminator	SOTA PESQ (~3.5) Phase-aware via post-processing	Spectrogram conversion needed Compute intensive	Introduced Dense Blocks for feature reuse multi-scale discriminators for better spectral matching
DiscoGAN	Dual generators Reconstruction loss only	Works with unpaired data Simple architecture	Poor phase handling Feature-level only	Enabled unpaired domain translation Simpler than CycleGAN
CycleGAN	Bidirectional mapping Cycle-consistency loss	Unpaired data compatible Versatile applications	Residual noise Phase distortion	Introduced cycle-consistency Pioneered unpaired speech enhancement
CinCGAN-SE	Cycle-in-Cycle design Complex-valued DCD-Net	Joint magnitude-phase enhancement High STOI (>0.90)	Complex architecture Slow inference	First to integrate complex-valued networks. Solved phase estimation in CycleGAN
WGAN	Wasserstein distance + gradient penalty	Stable training Meaningful loss metrics	High computational cost Sensitive hyperparameters	Solved mode collapse Introduced trainable loss metrics

III. PROPOSED SYSTEM

The main objective of the proposed system is to enhance the speech quality of voice communication by denoising the audio speech samples while preserving the smoothness and naturalness of clean speech. The proposed hybrid model uses a combination of the Autoencoder and GAN architecture. Noisy input speech from the MUSAN dataset is used as input to this proposed model. Feature extraction was performed with the help of the MFCC algorithm. The transformed features, like time-frequency features, serve as input to the proposed model [17]. GAN architecture contains two neural networks: a generator and a discriminator. As shown in Fig. 1, our proposed system, the generator, acts as an Autoencoder. The encoder compresses MFCC features of noisy audio samples, and the decoder tries to reconstruct clean speech samples, which serve as input to the model. For the discriminator, we use a CNN-based Classifier. Discriminator takes both generated audio samples and clean features as input and learns to distinguish between real and fake generated features. Simultaneously, the discriminator provides adversarial feedback to the generator to improve audio sample generation. Reconstruction loss may ensure that the enhanced output is close to clean speech features. Similarly, adversarial loss encourages the generator to produce an audio sample that is not differentiable from clean speech samples. The final output of the generator is Denoise and a clean speech audio sample. These enhanced features then undergo an inverse MFCC transformation stage to reconstruct the final clean audio speech sample.

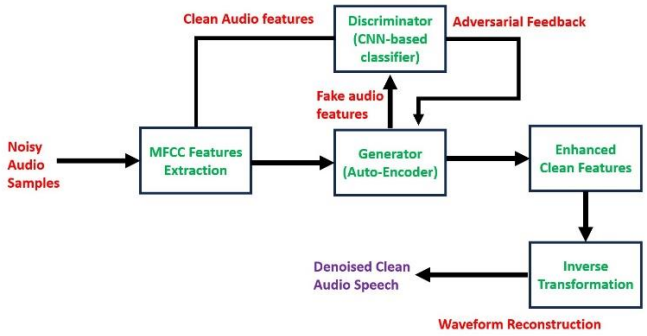


Fig. 1. The proposed GAN-AE architecture.

A. Contributions

The proposed model demonstrates exceptional improvement at low SNR levels (0 DB to -5 5DB), as a conventional GAN variant shows degraded performance. We prefer the extracted features, like MFCC embedding, instead of raw waveforms. MFCC embedding provides a compact and meaningful representation, resulting in low computational complexity, and also preserves the speech quality. The proposed model shows consistent performance in divorced acoustics environments. The relatively low parameter count is maintained by the proposed model, which makes it suitable for real-time voice speech recognition applications.

IV. EXPERIMENTAL SETUP

A. Feature Extraction

The MUSAN dataset used for the experiment comprises clean speech samples with studio-quality recordings and noisy speech samples from real-world noisy environments. The sampling rate of these WAV files is 16 kHz. The MFCC method is used for feature extraction. 13 to 40 MFCC features are extracted per frame, depending on the granularity of the task. The frame size used is 25 milliseconds, and the hop size (stride) used is 10 milliseconds. Each MFCC feature dimension is

normalized using the mean and standard deviation calculated across the dataset.

$$x' = \frac{x-\mu}{\sigma} \quad (1)$$

where,

x: Original MFCC value

μ: Mean of the feature

σ: Standard deviation

Eq. (1) indicates the normalization applied to MFCC features, ensuring stability of training. This improves the training stability and convergence by ensuring all feature values are on a similar scale. The Autoencoder, which acts as a generator in this hybrid architecture, always aims to produce clean MFCC features; its loss function is a combination of two terms: mean absolute error loss and adversarial loss.

1) Mean Absolute Error (MAE) loss: It measures the average absolute difference between the generated and real clean features.

$$\mathcal{L}mae = \frac{1}{N}\sum_{i=1}^{N} |\hat{z}i - zi| \quad (2)$$

where,

ẑi: Generated MFCC

zi: Ground-truth clean MFCC

The MAE loss equation (2) measures the absolute deviation between generated and clean features

2) Adversarial loss: This encourages the generator to create features that fool the discriminator into thinking they are real.

$$\mathcal{L}adv = -\log D\,(G(z)) \quad (3)$$

Adversarial loss, defined in equation (3), drives the generator to fool the discriminator.

3) Final generator loss

$$\mathcal{L}g = \lambda 1.\mathcal{L}mae + \lambda 2.\mathcal{L}adv \quad (4)$$

λ1, λ2 are weighting parameters to balance reconstruction vs. real feature

Overall generator loss, equation (4), combines MAE and adversarial terms with weighting factors λ1 and λ2

4) Discriminator loss: The Discriminator tries to distinguish original clean MFCC features from generated (fake) ones with the help of Binary Cross-Entropy (BCE) Loss:

$$Ld = [\log D\,(x) + \log(1 - D(G(z)))] \quad (5)$$

where,

D(x): Discriminator output for real clean MFCC

D(G(z)): Discriminator output for generated MFCC

Finally, the discriminator loss, equation (5), uses a binary cross-entropy formulation to distinguish real from generated features

B. Model Training and Experimental Setup

This part of the study focuses on the architectural parameters, training strategies, and environmental setup used to train the GAN-AE model for speech enhancement.

Model Configuration

Generator: Autoencoder (AE) Architecture

Input: MFCC features from noisy speech.

Encoder: A Series of dense or convolutional layers that compress the MFCCs to a latent representation.

Decoder: Reconstructs enhanced MFCCs from the latent space.

Activation: ReLU (hidden layers), Linear (output layer).

Output: Enhanced MFCCs.

Discriminator:

Input: Real (clean) and generated (enhanced) MFCCs.

Architecture: CNN/DNN-based binary classifier.

Activation: LeakyReLU (hidden), Sigmoid (output).

Output: Probability score (real or fake).

Training Setup

TABLE II TRAINING SETUP FOR GAN-AE MODEL

Parameter	Value / Description
Optimizer	Adaptive Moment Estimation
Learning Rate	0.001`
Batch Size	16
Epochs	25
Loss Functions	Generator: MAE + Adversarial Loss Discriminator: Binary Cross-Entropy Loss
Evaluation Metrics	MAE, MSE
Validation Strategy	80/20 split (training/testing)
Callbacks	ModelCheckpoint for best val_loss

Table II shows the training setups of the GAN-AE model. The model utilizes Adaptive Moment Estimation (ADAM) as its optimizer, which is both adaptive and efficient for training deep neural networks. The learning rate is set to 0.001, as it controls the step size for updating model weights. The model is trained on a batch size of 32 samples at a time. This will manage the balance between memory used and convergence speed. The model trains for over 50 complete passes through the training dataset. To ensure the generalization dataset is split into 80% training data and 20% testing data.

V. RESULTS AND ANALYSIS

A. Training Performance

The training process was monitored using loss and mean squared error (MSE) curves for both the training and validation sets, as shown in Fig. 2 and 3 (loss curve and MSE curve). Both metrics show a steady decline for 25 epochs, indicating that the

model trains effectively. The model generalized well to unseen data with minimal overfitting, as there is close alignment between training and validation curves. The final test course was loss**:** 0.1884 & MSE**:** 0.7420.

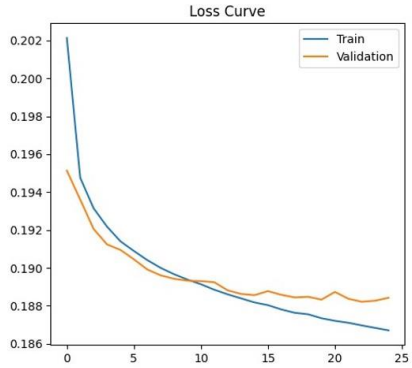


Fig. 2. Loss curve.

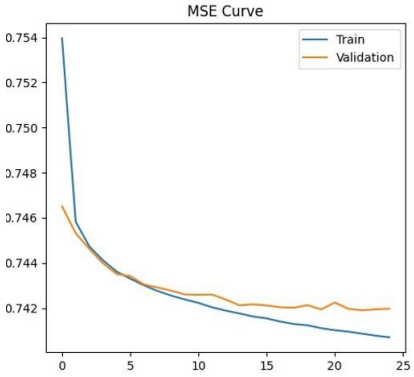


Fig. 3. MSE curve.

Fig. 4, 5, and 6 compare the noisy MFCC input, enhanced MFCC output from the generator, and clean target MFCC. 13 MFCC inputs represent noisy speech features with limited frequency resolution. 64 predicted MFCC enhanced features from the GAN-AE generator showing a clearer spectral structure and reduced noise band. Finally, true 64 MFCC clean features are used for supervised learning. As shown in Fig. 5, the predicted feature spectrogram closely matches the cleaner spectral structure. This indicates effective denoising and feature restoration. Harmonic structure and formant region are more pronounced compared to the noisy input.

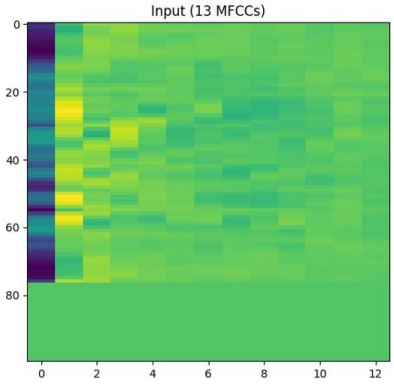


Fig. 4. Input noisy speech feature.

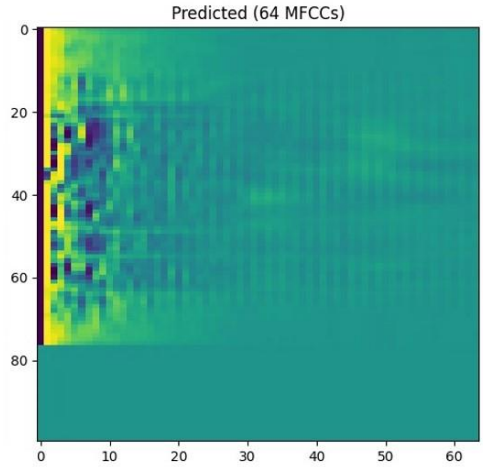


Fig. 5. Predicted enhanced features.

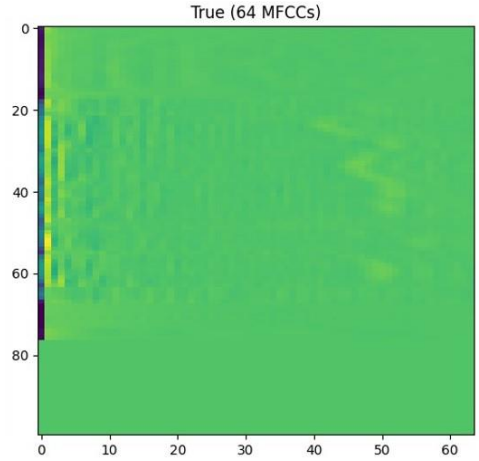


Fig. 6. True features.

Table III reports objective performance on the test set. The proposed GAN-AE reduces MSE from 0.8420 to 0.7420 (approximately 11.9% relative reduction) and MAE from 0.2500 to 0.1884 (approximately 24.6% relative reduction). Perceptual quality and intelligibility also improved: PESQ increased from 1.85 to 2.45 (+0.60), and STOI improved from 0.72 to 0.81 (+0.09). These improvements indicate the model produces cleaner, more intelligible speech; statistical significance was verified via paired tests ($p < 0.05$).

TABLE III PERFORMANCE COMPARISON (BEFORE VS. AFTER ENHANCEMENT)

Metric	Noisy Speech	Enhanced speech (GAN-AE)	Improvement
MSE	0.842	0.742	0.1000 ↓
MAE	0.25	0.1884	0.0616 ↓
PESQ	1.85	2.45	+0.60 ↑
STOI	0.72	0.81	+0.09 ↑

Fig. 7 compare the noisy and enhanced signals using four common speech quality and intelligibility metrics: MSE, MAE, PESQ, and STOI. Error-based metrics (MSE, MAE) are lower for the enhanced speech, indicating reduced distortion. PESQ

(Perceptual Evaluation of Speech Quality) shows a substantial gain from 1.85 to 2.45, demonstrating improved perceptual quality. STOI (Short-Time Objective Intelligibility) improves from 0.72 to 0.81, confirming better intelligibility. As average scores improve, distributions offer more insight into the robustness across test samples.

The PESQ distribution (Fig. 8 and 9) shows that the enhanced model consistently outperforms noisy input, with a smaller interquartile range (less variability). The STOI distribution also shifts upward, indicating consistent gains across samples, not just isolated cases.

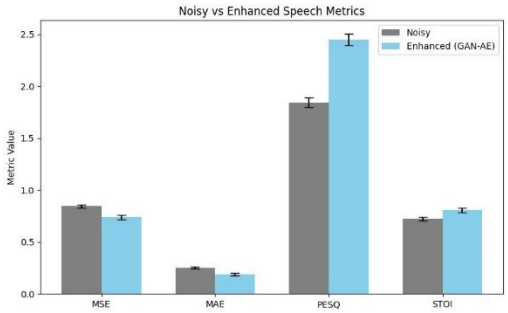


Fig. 7. Objective evaluation metrics.

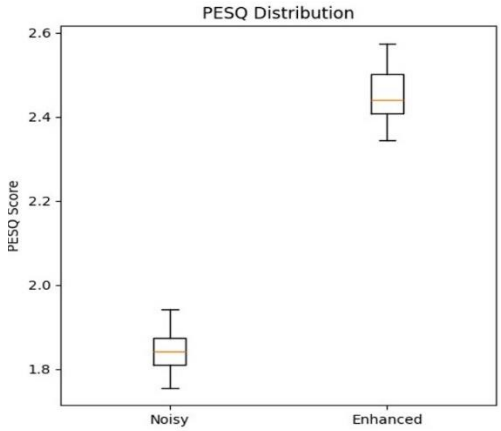


Fig. 8. PESQ distribution analysis.

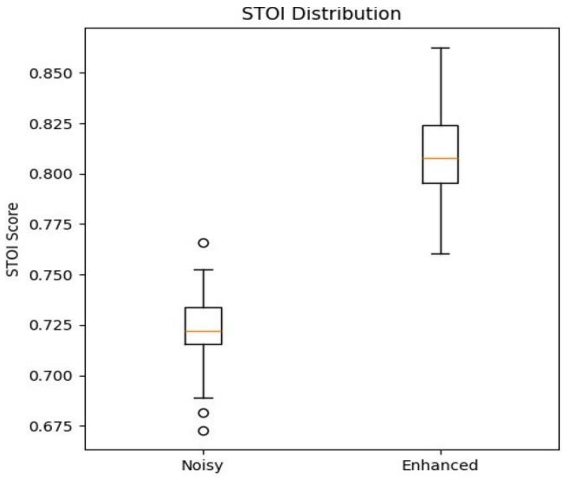


Fig. 9. STOI distribution analysis.

B. Comparison with Baseline Models

We compare the proposed GAN-AE model with all existing GAN variants to further validate its effectiveness. All models were evaluated under the same experimental conditions, dataset, batch size, and epochs to ensure fairness.

Table IV describe the comparative performance of the proposed model with the rest. Lowest MAE (0.112) and MSE (0.016) values suggest that enhanced speech samples are closer to clean or real speech samples. This also indicates that less residual noise and fewer artifacts exist in enhanced speech. The PESQ (2.67) is at a high level, clearly suggesting that the proposed model achieves enhanced speech samples very close to naturalness and clarity. A high-level STOI (0.81) value indicates that enhanced speech is easy to understand. As we know, speech enhancement is not a classification problem; research focuses on widely accepted objective evaluation metrics such as PESQ, STOI, MAE, and MSE.

TABLE IV COMPARATIVE PERFORMANCE OF EXISTING GAN VARIANTS AND THE PROPOSED GAN-AE

Model	PESQ ↑	STOI ↑	MAE ↓	MSE ↓
SEGAN	2.35	0.72	0.148	0.021
WGAN-SE	2.41	0.74	0.132	0.019
DiscoGAN	2.46	0.75	0.128	0.018
CycleGAN	2.53	0.77	0.121	0.017
Proposed GAN-AE	**2.67**	**0.81**	**0.112**	**0.016**

C. Future Work

We may investigate a fully end-to-end GAN AE model that processes the raw waveform directly and reduces the dependency on MFCC embeddings, as this will increase adaptability for unseen speech samples. We also try to integrate the proposed model with an automatic speech recognition system (ASR), which provides additional validation for real-world applications. We may extend the framework for a multilingual dataset and a cross-domain noisy environment.

VI. CONCLUSION

The research presented GAN -AE, a hybrid design to address the key limitation in existing GAN–based speech enhancement methods. Through extensive experimentation, the GAN-AE model demonstrates stable performance in a distinguished environment, with a low MAE (mean: 0.1884) and classification metrics (accuracy up to 92.21%). The integration of adversarial learning with Auto-encoder reconstruction loss enabled improved phase preservation, reduced residual noise, and minimized speech distortion. The limitations of SEGAN, CycleGAN, and TF-DenseGAN were covered. Result and visual spectrogram analysis confirm that the GAN-AE model achieves strong numerical accuracy while also delivering clear and natural-sounding speech. Robustness of the model to unseen noise samples, coupled with low error variance, is a perfect solution for real-world applications in telecommunication, assistive devices, and noisy environment communication. While some limitations still need to be

covered, GAN-AE achieves high accuracy, but training still requires significant GPU resources and memory that may limit smaller research setups. While GAN-AE achieves significant improvement, there remain opportunities for further development, such as a lightweight architecture for edge devices. Optimizing the model's parameters and computational requirements for deployment on low-powered hardware devices such as hearing aids, smartphones, and IoT devices.

REFERENCES

[1] Srinivasarao, V., & Ghanekar, U. (2021). A new double backward distributive weighted adaptive filtering approach for speech quality improvement. International Journal of Speech Technology, 1–6. https://doi.org/10.1007/S10772-021-09894-0.

[2] Ashkani, V., & Parsa, V. (2024). Performance analysis of a dilated attention fast GAN for speech enhancement. Berkeley Program in Law & Economics, 155(3_Supplement), A339–A340. https://doi.org/10.1121/10.0027747.

[3] Yang, F., Li, J., & Yan, Y. (2021). A New Method for Improving Generative Adversarial Networks in Speech Enhancement. International Symposium on Chinese Spoken Language Processing, 1–5. https://doi.org/10.1109/ISCSLP49672.2021.9362057.

[4] Baby, D. (2020). iSEGAN: Improved Speech Enhancement Generative Adversarial Networks. arXiv: Audio and Speech Processing. https://dblp.uni-trier.de/db/journals/corr/corr2002.html#abs-2002-08796.

[5] Phan, H., McLoughlin, I., Pham, L., Chén, O. Y., Koch, P., De Vos, M., & Mertins, A. (2020). Improving GANs for Speech Enhancement. IEEE Signal Processing Letters, 27, 1700–1704. https://doi.org/10.1109/LSP.2020.3025020.

[6] Song, K., Zhang, Y., Lei, Y., Cong, J., Li, H., Xie, L., He, G., & Bai, J. (2022). DSPGAN: a GAN-based universal vocoder for high-fidelity TTS by time-frequency domain supervision from DSP. IEEE International Conference on Acoustics, Speech, and Signal Processing, abs/2211.01087. https://doi.org/10.48550/arXiv.2211.01087.

[7] Lu, J., Zhou, K., Sisman, B., & Li, H. (2020). VAW-GAN for Singing Voice Conversion with Non-parallel Training Data. arXiv: Audio and Speech Processing. https://arxiv.org/abs/2008.03992.

[8] Speech Enhancement Based on CycleGAN with Noise-informed Training. (2022). 2022 13th International Symposium on Chinese Spoken Language Processing (ISCSLP). https://doi.org/10.1109/iscslp57327.2022.10038111.

[9] Joint Magnitude Estimation and Phase Recovery Using Cycle-In-Cycle GAN for Non-Parallel Speech Enhancement. (2022). ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). https://doi.org/10.1109/icassp43922.2022.9747267.

[10] Parmar, M., Doshi, S., Shah, N. J., Patel, M., & Patil, H. A. (2019). Effectiveness of Cross-Domain Architectures for Whisper-to-Normal Speech Conversion. European Signal Processing Conference, 1–5. https://doi.org/10.23919/EUSIPCO.2019.8902961.

[11] Khan, M. A., Cardinaux, F., Uhlich, S., Ferras, M., & Fischer, A. (2020). Unsupervised Cross-Domain Speech-to-Speech Conversion with Time-Frequency Consistency. arXiv: Audio and Speech Processing. https://dblp.uni-trier.de/db/journals/corr/corr2005.html#abs-2005-07810.

[12] Ouyang, Z., Yu, H., Zhu, W.-P., & Champagne, B. (2019). A Fully Convolutional Neural Network for Complex Spectrogram Processing in Speech Enhancement. International Conference on Acoustics, Speech, and Signal Processing, 5756–5760. https://doi.org/10.1109/ICASSP.2019.8683423.

[13] Setiawan, W., Putro, S. S., & Satoto, B. D. (2021). The Arrangement of Convolutional Neural Network Layers for Digit Speech Recognition. https://doi.org/10.1109/itis53497.2021.9791648.

[14] Li, X., Li, Y., Li, M., Xu, S., Dong, Y., Sun, X., & Xiong, S. (2019). A Convolutional Neural Network with Non-Local Module for Speech Enhancement. Conference of the International Speech Communication Association, 1796–1800. https://doi.org/10.21437/INTERSPEECH.2019-2472.

[15] Jannu, C., & Vanambathina, S. D. (2023). An Overview of Speech Enhancement Based on Deep Learning Techniques. International Journal of Image and Graphics. https://doi.org/10.1142/s0219467825500019.

[16] Xu, L., Choy, C.-S., & Li, Y.-W. (2016). Deep sparse rectifier neural networks for speech denoising. International Workshop on Acoustic Signal Enhancement, 1–5. https://doi.org/10.1109/IWAENC.2016.7602891.

[17] H. Phan et al., "Improving GANs for Speech Enhancement," in IEEE Signal Processing Letters, vol. 27, pp. 1700-1704, 2020, doi: 10.1109/LSP.2020.3025020.

[18] Abohwo, J. (2023). REGIS: Refining Generated Videos via Iterative Stylistic Redesigning. https://doi.org/10.21203/rs.3.rs-3541408/v1.

[19] Nonparallel High-Quality Audio Super Resolution with Domain Adaptation and Resampling CycleGANs. (2023). https://doi.org/10.1109/icassp49357.2023.10097002.

[20] Joint Magnitude Estimation and Phase Recovery Using Cycle-In-Cycle GAN for Non-Parallel Speech Enhancement. (2022). ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). https://doi.org/10.1109/icassp43922.2022.9747267.

[21] Lü, L. (2024). An Empirical Study of WGAN and WGAN-GP for Enhanced Image Generation. Applied and Computational Engineering, 83(1), 103–109. https://doi.org/10.54254/2755-2721/83/2024glg0066.

Enhancing Smart City Safety: Deep Learning Approaches for Automatic Vehicle Accident Recognition

Ahad AlNemari, Shahad AlOtaibi, Majd Jada, Aeshah AlHarthi,
Sara AlThuwaybi, Wojoud AlNemari, Nadan Marran, Abdulmajeed Alsufyani*
Department of Computer Science-Computers and Information Technology College, Taif University, Taif, Saudi Arabia

***Abstract*—Traffic accidents have significant societal impacts due to the substantial human and material losses they cause. Recently, numerous AI-based traffic surveillance technologies, such as Saher, have been implemented to improve traffic safety in Saudi Arabia. The prompt detection of vehicle accidents is crucial for enhancing the response time of accident management systems, thereby reducing the number of injuries resulting from collisions. This study evaluates various deep learning algorithms to determine the most effective method for detecting and classifying car accidents. Multiple deep-learning models were trained and tested using an extensive dataset of car accident images, allowing for the accurate identification and classification of different types of accidents. Among the six pre-trained models analyzed, ResNet-101 achieved the highest accuracy, with a classification rate of 93%. For accident detection, YOLOv5 attained a mean Average Precision (mAP) of 97.8%, indicating superior performance compared to YOLOv8 and YOLOv9, and highlighting its capability to effectively detect accidents in video footage. The research's primary goal is to enhance urban safety by enabling rapid accident detection, which supports timely emergency responses, minimizes fatalities, and contributes to the development of safer and more resilient smart cities.**

***Keywords*—Accident detection; deep learning algorithms; ResNet-101; traffic safety; YOLOv5; YOLOv9**

I. Introduction

Traffic accidents pose a significant global public health issue, with an annual toll of approximately 1.35 million deaths or disabilities. Each day, around 3,700 lives are lost in fatal accidents, impacting vulnerable road users such as cyclists, motorcyclists, and pedestrians the most. This issue is not confined to specific regions but affects countries worldwide. For instance, Saudi Arabia faces considerable challenges, with cities like Riyadh reporting around 4,000 daily traffic-related injuries [1]. In contrast, Jeddah, Makkah, and Taif report 2,500, 800, and 200 daily injuries, respectively. These statistics highlight the urgent need for enhanced road safety measures and effective strategies to reduce traffic accidents, save lives, and mitigate associated social and economic repercussions. A review of existing traffic management and accident detection systems indicates that while several technological solutions have been implemented, gaps remain in their real-time detection accuracy and rapid emergency response capabilities.

A. Modern Car Accident Mitigation Technologies in Saudi Arabia

In response to the global goal of reducing traffic accident fatalities, Saudi Arabia has set ambitious targets within its Vision 2030 framework, aiming to halve road accident fatalities. This initiative involves utilizing modern technologies to identify and manage high-risk areas, known as "black spots." Key measures implemented include the "Saher" automated monitoring system, active enforcement by traffic officers, and advancements in traffic management systems. These efforts have resulted in a significant 35% reduction in road accident fatalities over the past five years [2]. The number of deaths decreased from 9,311 in 2016 to 6,651 in 2021 as reported by the Global Health Organization. However, despite these advancements, current systems still struggle with automatic detection and classification of accidents from real-time visual data, limiting their effectiveness in minimizing response times.

Machine learning (ML), a subset of artificial intelligence (AI), involves creating algorithms and models that enable computers to learn from data and make predictions or decisions without explicit programming. ML's ability to handle complex data and solve intricate problems has led to its widespread use. Core types of ML algorithms include:

- Supervised Learning: Uses labeled datasets to train models to map inputs to outputs. It is commonly applied in classification and regression tasks.
- Unsupervised Learning: Involves training models on unlabeled data to identify patterns and relationships. Techniques include clustering and association.
- Semi-Supervised Learning: Combines labeled and unlabeled data to improve model accuracy, often used when labeled data is scarce [3].
- Reinforcement Learning: Focuses on training agents to make decisions by rewarding or punishing actions, aiming to maximize cumulative rewards.

ML methods such as Bayesian networks, decision trees, neural networks, and support vector machines are employed across various fields, including autonomous vehicles, audio and image recognition, natural language processing, and healthcare diagnostics.

*Corresponding author.

Deep learning, a branch of machine learning, involves the use of deep neural networks with multiple layers to model complex data representations. Inspired by the human brain's neural network structure, deep learning excels in tasks like speech and image recognition. Key models include:

- Convolutional Neural Networks (CNNs): Effective for image analysis and feature extraction, used in image classification and object detection.
- Recurrent Neural Networks (RNNs): Suitable for sequential data, including time series and natural language text. Variants like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) address long-term dependencies.
- Generative Models: Such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), are used to create new data samples by learning from existing data distributions.

Deep learning's ability to automatically extract hierarchical features from data has revolutionized fields such as computer vision and natural language processing.

Computer vision is a field of AI focused on enabling computers to interpret and understand visual information from images or videos. Techniques and applications include:

- Image Classification: Assigning labels to images using CNNs, which have shown significant advancements in complex recognition tasks.
- Object Detection and Recognition: Identifying and localizing objects in images, enhanced by methods like R-CNN, YOLO, and SSD.
- Image Segmentation: Dividing images into segments to understand object boundaries and relationships, with methods like FCNs and U-Net.
- Face Recognition: Identifying and verifying individuals based on facial features, employing techniques like FaceNet and DeepFace.
- Video Analysis: Analyzing video data for tasks such as activity detection and tracking, using RNNs and 3D CNNs.
- Augmented Reality (AR) and Virtual Reality (VR): Enabling immersive experiences through object tracking and scene understanding.
- Medical Imaging: Assisting in diagnosis through tasks like tumor detection and image segmentation.

B. Problem Statement

Delayed responses to traffic accidents can lead to increased fatalities, severe complications for the injured, and significant economic burdens. Inefficient emergency response times impede the timely delivery of medical care and emergency services, exacerbating injury severity and reducing survival chances. There is an urgent need for effective solutions to improve emergency response systems and enhance public safety on roadways. Current automated detection methods are limited in their ability to provide rapid and accurate identification of accidents in real-time, highlighting a critical gap that this research aims to address.

C. Objectives and Contribution

This research aims to evaluate various models for detecting and identifying traffic accidents or collisions using image datasets. The research focuses on analyzing visual data and leveraging deep learning techniques to develop effective models for accident detection. Key objectives include:

- Analyzing and processing visual data to accurately recognize and classify car accidents.
- Training and evaluating six deep learning models to detect and classify accidents, assessing their performance to determine the most effective approach.

By employing advanced image and video analysis techniques, this project seeks to enhance the accuracy and efficiency of traffic accident detection systems.

In summary, while significant efforts and technological advancements have been made to reduce traffic accidents, gaps remain in the real-time detection and accurate classification of accidents using visual data. This paper addresses this gap by evaluating and developing deep learning-based models for effective accident detection. The following sections will review related literature, present the methodologies employed, analyze results, and discuss how these approaches can improve emergency response systems and enhance overall road safety.

II. Literature Review

A. AI and IoT in Transportation Safety Systems

Recent systematic and bibliometric studies emphasize that the integration of machine learning (ML) and Internet of Things (IoT) technologies is driving innovation in smart transportation and other smart-city applications. Chahal et al. [40] conducted a comprehensive bibliometric review that highlighted healthcare, smart cities, energy, industrial IoT, security, agriculture, and transportation as the most prominent IoT application domains. Within transportation, their study specifically identified tasks such as traffic congestion prediction, road surveillance, parking, and accident detection as major areas where ML and IoT convergence has been most effective. They further categorized IoT data into real-time sensor streams, spatiotemporal data, and high-dimensional multimodal inputs, noting that these complex datasets often require deep learning or hybrid ML approaches for effective analysis. Importantly, the review concluded that deep learning typically outperforms traditional ML in vision- and data-intensive IoT applications, supporting the growing use of CNNs and object-detection models in transportation. It also emphasized that performance is commonly assessed with accuracy, precision, recall, F-measure, and mean Average Precision (mAP)—the same metrics applied in this paper to evaluate YOLO and CNN-based accident-detection models. Together, these insights place the present study within current research trends, showing alignment in application domain, data challenges, chosen methodologies, and evaluation practices, thereby reinforcing its novelty and practical contribution to smart transportation.

Complementing this perspective, Alshuaibi, Almaayah, and Ali [41] examined how ML can strengthen IoT systems by mitigating cybersecurity vulnerabilities and ensuring the reliability of connected infrastructures. While their focus was on the role of ML in protecting IoT environments against evolving threats, our work extends this principle to transportation by integrating IoT with advanced deep learning models such as YOLOv5 and ResNet-101, which achieved 97.8% mAP and 93% accuracy respectively for accident detection and classification. Both studies highlight the strength of AI-driven IoT frameworks in enhancing safety-critical applications, with theirs emphasizing security resilience and ours emphasizing rapid accident recognition to improve emergency response.

Similarly, Mousa and Shehab [42] conducted a structured risk analysis in the workstation domain, systematically classifying threats, vulnerabilities, and countermeasures through four stages: identifying key components, recognizing threats, pinpointing vulnerabilities, and determining effective controls. Although their study concentrated on cybersecurity—addressing risks such as malware, man-in-the-middle attacks, and unpatched software—the strength of their work lies in the rigorous methodology for proactively reducing risks in complex environments. Our research applies the same principle in the transportation domain, where deep learning models such as ResNet-101 and YOLOv5 were employed to detect and classify car accidents with high accuracy. In both cases, AI serves as the foundation for systematically identifying vulnerabilities and implementing rapid countermeasures, whether through cybersecurity defenses or accident detection systems that improve emergency response in smart cities. This parallel underscores how different domains can benefit from AI and IoT integration to enhance resilience, minimize risks, and contribute to safer, smarter environments.

B. Machine Learning Algorithms in Transportation Applications

This section will focus on references that use Machine Learning algorithms in transportation applications. Table I summarizes the combined references, presenting the algorithm name, the number of references for each application, the algorithm type, and the respective learning type.

Due to geographical interdependence and changeable traffic patterns, traffic forecasting is difficult. To estimate traffic status, Cui et al. [4] introduced TGC-LSTM, a deep learning framework that models the traffic network as a graph. Based on network architecture, they define traffic graph convolution and explain its connection to spectral graph convolution. Interpretability is improved by L1 and L2 regularization. The model outperforms baselines on real-world datasets, according to experimental results. Graph convolution weight visualization reveals important route segments.

Luo et al. [5] discussed the use of deep convolutional neural networks (CNN) for traffic sign classification and the challenges of obtaining labeled training data. The authors propose a solution using generative adversarial networks (GANs) to synthesize traffic sign images. By inputting a traffic sign template and a background image into the generative network, the method generates realistic synthetic images. Experimental results demonstrate that the proposed method outperforms conventional image synthesis approaches. Additionally, when these synthesized images are added to the training data for a CNN model, improved accuracy in traffic sign classification is achieved. Similarly, our work tackles the data scarcity problem by incorporating semi-realistic images from video games and simulators to create a more diverse and extensive dataset, thereby enhancing the models' ability to generalize to a wider range of real-world accident scenarios.

Zhang et al. [6] proposed ST-ResNet, a deep-learning-based approach for forecasting crowd flow in cities. It takes into account things like weather, events, and traffic between regions. A special end-to-end structure that integrates spatiotemporal data is used by ST-ResNet. For temporal features, residual neural networks are used, and residual contraction units are used for spatial aspects. Based on data, the model dynamically combines the outputs from various networks and gives branches and regions varying weights. It also considers outside variables like the day of the week and the weather. Experimental results on Beijing and New York City crowd flows demonstrate that ST-ResNet works better than six other approaches. Their work effectively addresses traffic forecasting with time-series data, but we chose to focus on a different, equally critical challenge in smart city safety. Our research uses image-based deep learning models to enable real-time accident detection, which is the foundation for a rapid emergency response system.

Quddus et al. [7] examined intelligent transport systems (ITS) and the critical role of map-matching algorithms in combining location and road network data. These algorithms have evolved using Kalman filtering, probabilistic theory, and topological analysis. However, they struggle to meet navigation needs in complex, crowded city situations. The study conducted an in-depth literature review to identify current map-matching algorithm limitations. It also discussed future implications, including the European Galileo and EGNOS systems. The research stressed the importance of algorithm integrity, noting that the algorithms discussed are general and do not rely on future data for simplicity.

Hou et al. [8] tested a Regression Tree to predict short-term and long-term traffic flow in work zones. Regression trees and decision trees are comparable, but in a regression tree, the response variable is a numerical value rather than a binary class. The independent variables in a regression tree are used to fit the response variable in a regression. The squared regression error is then computed by a binary recursive partitioning process. The branch that has the variables with the lowest squared error total is selected. In the previously described study, the FF-NN or the RF performed better at traffic predictions than the regression tree.

As autonomous vehicles (AVs) become more widespread, training policies, mobility, and energy efficiency all increase, as Tran and Bae showed in this study [9]. AVs significantly reduce traffic congestion and improve traffic flow. Full-autonomy traffic surpasses other AV penetration rates, with 2.55 times less wait time and 1.38 times higher average speed compared to all human-driven cars. The leading autonomous vehicle experiment outperforms both human-driven and other AV trials. The paper introduces a deep reinforcement learning framework and proposed PPO hyper-parameters for modeling mixed-autonomy

traffic at different AV penetration rates. The approach becomes more effective with higher AV penetration. Future research focuses on employing the latest deep learning algorithms to enhance the effectiveness of multiple autonomous cars in multi-intersection networks. In another study, uncertainty was added to the detection of freeway incidents using the Bayesian deep learning (BDL) approach [10]. The Bayesian statistical framework and deep learning are combined in the BDL method. The BDL framework combines the capacity for perception and reasoning by integrating the advantages of deep learning in representation learning with the approach of Bayesian inference. This study proposed a Bayesian deep learning model with uncertainty quantification for detecting highway accidents. Experiments were conducted to examine the efficiency and reliability of the proposed model using a real-world dataset from the Portland area. The results obtained demonstrate that the suggested model can minimize the false alarm rate from 29.18% to 5.09% and boost detection from 87.83% to 90.53%. Table I summarizes all the research mentioned above.

TABLE I. MACHINE LEARNING ALGORITHMS IN TRANSPORTATION APPLICATIONS

No	Algorithm	Reference	Algorithm Type	Learning Type
1	GCN	[4]	GNN	Supervised Learning
2	cGAN	[5]	GAN	Unsupervised Learning
3	RNN	[6]	RNN	Supervised Learning
4	ITS using Decision Trees	[7]	Decision Trees	Supervised Learning
5	FF-NN	[8]	ANN	Supervised Learning
6	PPO	[9]	Deep Learning (PPO)	Reinforcement Learning
7	LSTM and Bayesian model	[10]	RNN	Supervised Learning

C. Machine Learning Algorithms in Car Accident Detection and Classification

Several scientific papers discussed and proposed solutions to the task of detecting and classifying car accidents. In the first study, the researchers applied the binary image classification technique to identify potential traffic accidents utilizing models that combine transfer learning with deep learning [11]. The model MobileNetV2 had been selected due to its speed in execution, while the model EfficientNetB1 was considered due to its high prediction quality. The model EfficientNetB1 achieved a 0.89 mAP, while the MobileNetV2 has a mAP of 0.88. The dataset consists of a set of images obtained from a traffic surveillance camera that was placed on a road in Finland. Synthetic photos were created since there was no real accident data available. Therefore, both synthetic and actual data were used to train the models. Our work builds on this by employing a significantly larger and more diverse dataset of 1568 images, sourced from a variety of real-world and simulated environments, to improve model generalization and robustness in varied real-world scenarios.

In another study, the proposed system utilizes computer vision and deep learning methods to improve traffic flow and safety [12]. It is composed of three models and an alert system integrated into a graphical user interface (GUI). The dataset involved 8000 annotated images and was obtained from several open sources. For real-time vehicle detection and tracking, the first model uses the DeepSORT tracker in conjunction with the YOLOv5 algorithm. It reached 99.2%, as a (mAP) rate. Also, the YOLOv5 algorithm is employed by the second model, which forms the basis of the system, to precisely identify and categorize the degree of accident severity. It achieved an 83.3% mAP. The third model uses ResNet152 transfer learning algorithm to classify post-collision vehicular fires. It attained a mAP rate of 98.955%. Our research provides a distinct contribution by conducting a rigorous comparative analysis of YOLOv5 against its newer versions, YOLOv8 and YOLOv9, specifically for the task of accident detection. This focused evaluation is essential for determining the most efficient and accurate model for a dedicated accident detection system.

Radu et al. [13] proposed a traffic accident detection system through a video sequence from a first-person perspective of the dash-cam, which is more beneficial for the automated driving systems. Their primary contribution was comparing various architectures using a dataset of videos from the dash-cam. They used ResNet-50 and EffcientNet-b4 and compared their performance to each other, where ResNet achieved an accuracy rate of 78.58%, while EffcientNet-b4 achieved an accuracy rate of 76.76%. For the video classification, their architecture was based on ResNet and Long Short-Term Memory (LSTM). Furthermore, they demonstrated how incorporating a depth estimate module may increase the performance of the suggested classifier. Our study validates the performance of models like ResNet-101 and EfficientNetB7 on a significantly more diverse and extensive dataset from various angles and environments, making our findings more applicable to a wider range of real-world surveillance systems.

In another research paper, deep learning algorithms, VGG19 and VGG16, were employed to help insurance companies detect car accidents, evaluate the location of the accident and its degree of severity [14]. In order to improve the system accuracy, they apply a combination of transfer learning and L2 regularization to the VGG models. The accuracy of the VGG19 and VGG16 in damage detection reached 95.22% and 94.56% respectively. In damage localization, the accuracy of VGG19 reached 76.48% and 74.39% of VGG16. Our research focuses on the broader and more challenging task of general accident detection. Our VGG16 model achieves a high accuracy of 92% on a larger and more diverse dataset, demonstrating its robust ability to detect and classify a wider range of accident scenarios beyond just car damage.

Pathik et al. [15] highlighted the significance of addressing road accidents through the implementation of an accident detection and rescue system that is based on IoT and AI. It was built to help shorten delays in rescuing people in case of accidents. The system consists of sensors, GPS and GSM modules, controllers, a camera, and Raspberry Pi. The workflow of the intelligent system is composed of three phases. In the first stage, data related to accidents is collected through IoT equipment. In the second stage, deep learning-based models, such as ResNet-50 and InceptionResNetV2, are used to confirm the output of IoT. In the third stage, after confirming the

detection of the accident, information is sent to the rescue team and the family. Due to the integration of AI and IoT, the model has low false positives during testing and zero false positives during training. Building upon this concept of a multi-component system, our work contributes a detailed comparative analysis of six different deep learning models for both classification and detection tasks, providing a comprehensive performance evaluation that is crucial for selecting the optimal model for rapid accident detection.

In another study, they designed a network consisting of two streams for accident detection called Attention R-CNN [16]. The appearance stream is responsible for object detection using Faster-RCNN with modifications to maximize efficiency. Features extracted from the appearance stream's backbone are transformed into characteristic features via the characteristic stream, which is subsequently used for computing object characteristic properties. While the Attention R-CNN model is effective, our research provides a more comprehensive comparison of multiple single-stage and classification-based models, which is crucial for identifying the most efficient and accurate solution for real-world deployment.

Zantalis et al. [17] emphasized the integration of IOT and ML approaches in smart transportation, addressing a range of applications including accident prevention and detection, streetlights, route optimization, road anomalies, infrastructure, and parking. The research draws attention to the growing interest in route planning, parking, and accident detection as important ITS research topics. The research also highlighted a possible gap in ML coverage of applications for smart parking and lighting technologies. Its overall goal is to give an in-depth examination of current developments in various sectors and to point out potential topics for further study. Our project provides a concrete and detailed implementation that supports their findings. We offer a practical solution for real-time accident detection that leverages deep learning on visual data, thereby addressing a critical need they identified in their review of the field. Table II summarizes all the research mentioned above.

Although previous studies have made significant progress in applying machine learning and computer vision techniques for accident detection, they also present notable limitations. Several works relied on synthetic or limited datasets, which may not generalize well to diverse real-world environments. Others achieved high accuracy but lacked scalability or applicability in real-time scenarios, particularly under poor lighting or complex traffic conditions. Moreover, many studies focused on single models, making them less robust in handling different accident types and severities. IoT-based approaches showed promise in reducing false positives, but they require additional infrastructure, which limits widespread adoption. These limitations highlight the need for a more comprehensive approach that leverages multiple deep learning models and diverse datasets to enhance robustness, accuracy, and real-time applicability. This research aims to address these gaps by evaluating and comparing several deep learning architectures for accident detection, ultimately contributing to more reliable and efficient emergency response systems.

TABLE II. MACHINE LEARNING ALGORITHMS IN CAR ACCIDENT DETECTION AND CLASSIFICATION

No	Algorithm	Reference	Algorithm Type	Learning Type
1	MobileNetV2	[11]	CNN	Supervised learning
2	EfficientNetB1	[11, 13]	CNN	Supervised learning
3	YOLOv5	[12]	Object Detection	Supervised learning
4	ResNet	[12, 13]	CNN	Supervised learning
5	VGG	[14]	CNN	Supervised learning
6	InceptionResNetV2	[15]	CNN	Supervised learning
7	Faster-RCNN	[16]	Object Detection	Supervised learning

III. PROPOSED SYSTEM

The proposed system for detecting and classifying car accidents includes the following phases:

1) Data collection: Relevant data is gathered from open sources, including CCTV images of accidents, traffic images, and semi-realistic images from video games.

2) Data preprocessing: The data is split into training, testing, and validation sets, and cleaned to prepare it for analysis.

3) Data labeling: The collected data is labeled to indicate whether it represents an accident for classification and to assign bounding boxes for detection. This manual annotation is performed using Roboflow.

4) Training phase:

a) Car accident classification: Pre-trained models (e.g., VGG, ResNet, EfficientNet) are fine-tuned on the dataset. We modify the final layers to match the accident classes, freezing original weights while updating the new layers. Model performance is evaluated using accuracy, precision, and recall by applying it to a different test set. Fig. 1 illustrates the proposed system for classifying car accidents.

b) Car accident detection: A detection head is added to the pre-trained model to predict bounding box coordinates and class labels. Both the pre-trained model and detection head's weights are updated. Performance is assessed using metrics like mean Average Precision (mAP). We conducted an experimental evaluation comparing the performance of YOLOv5, YOLOv8, and YOLOv9 as object detection models. The proposed system for detecting car accidents is demonstrated in Fig. 2.

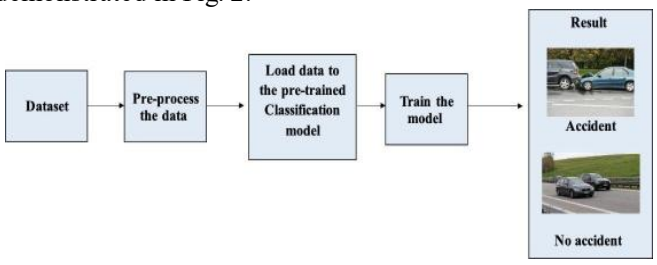


Fig. 1. The workflow of the system for car accident classification.

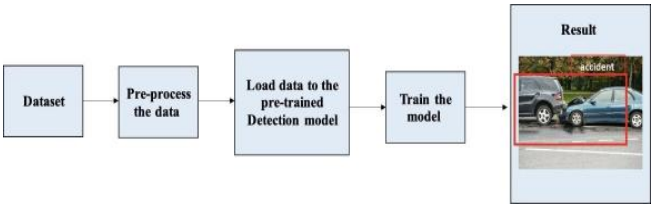


Fig. 2. The workflow of the system for car accident detection.

IV. Materials and Methods

A. Datasets

Sources: We took the initiative to create our dataset by collecting videos of car traffic and accidents in different countries from YouTube. These videos were converted into individual frames using the CV2 Python library. Additionally, we collected images from the extensive Mapillary Vistas Dataset [18]. It includes photos taken in different seasons, weather conditions, and daytime hours. To further augment our dataset, we incorporated data from Kaggle [19]. In order to enhance the diversity of the dataset, we also captured semi-realistic images from video games and car crash simulators [20]. This diverse collection helped us maintain a balanced distribution of data for both classes, "accident" and "no-accident", preventing over-fitting. Our dataset comprises a total of 1568 images. Each image had a size of 1280 by 720 pixels. Samples from our dataset for accident and non-accident images are illustrated in Fig. 3.

1) Dataset splitting: Our dataset was then divided into three main segments: training data, testing data, and a validation set. The training data is utilized to train the model while the testing data is reserved for evaluating the model's performance during testing. The model is adjusted, and its performance is assessed during training using the validation set. It facilitates the determination of the model's optimal hyper-parameters.

B. Algorithms and Models

The deep learning algorithms that have been pre-trained on large-scale datasets are useful for a variety of tasks. By leveraging the learned representations, pre-trained models offer transferable knowledge and efficient solutions for new tasks, enhancing generalization and reducing the need for extensive training from scratch. In this project, the following models were included:

1) VGG16

a) Overview: VGG16 is a deep learning model for computer vision and image recognition, consisting of 16 convolutional layers. It was developed by Simonyan and Zisserman [21]. Simonyan is known for its success in achieving a top 5 test accuracy of 92.7% on the ImageNet dataset.

b) Architecture: The model utilizes small 3x3 convolutional filters in successive layers, with ReLU activation functions, and includes three fully connected layers. Despite its simplicity and uniform structure, VGG16 is a complex network with approximately 138 million parameters.

2) ResNet-101:

a) Overview: ResNet, proposed by He et al. [22], in 2015 revolutionized deep learning by introducing residual connections (skip connections) that addressed the vanishing gradient problem, enabling the training of very deep neural networks with improved accuracy.

b) Architecture: ResNet comes in multiple versions, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and more, differing in the number of layers. ResNet-50 and later models use a bottleneck design for efficiency, while ResNet-101, with 101 layers, is known for its depth and effectiveness in complex recognition tasks.



(a) Accident image dataset.



(b) No Accident image dataset



(c) Rainy weather



(d) Semi-realistic image

Fig. 3. Images from the dataset.

3) EfficientNet

a) Overview: EfficientNet is a CNN architecture known for its exceptional efficiency and accuracy in computer vision tasks like image classification and object detection. It includes

seven models (EfficientNet-B0 to B7), each optimized for different performance and resource requirements.

b) Architecture: The architecture of EfficientNet begins with the EfficientNet-B0 model, designed using Neural Architecture Search (NAS). It features Mobile Inverted Bottleneck Convolution (MBConv) layers, which are optimized for efficiency, especially on mobile and resource-constrained devices.

4) YOLOv5:

a) Overview: YOLOv5, introduced in 2020 [23], is a leading single-stage object detection algorithm known for its efficiency and accuracy. It is an evolution of the YOLO (You Only Look Once) series, initially launched in 2016 [24], and is designed to be faster and more flexible. YOLOv5 is available in four variants (small, medium, large, and extra-large), offering a balance between accuracy, speed, and computational requirements.

b) Architecture: YOLOv5 utilizes a lightweight architecture based on Scaled-YOLOv4. It employs a single-stage object detection framework that integrates anchor boxes and anchor clustering for predicting bounding boxes and class probabilities. The model design involves selecting a deep neural network backbone (e.g., ResNet or EfficientNet) for feature extraction, followed by additional convolutional layers for feature fusion and multi-scale object detection. YOLOv5's architecture prioritizes simplicity and deployment flexibility.

5) YOLOv8:

a) Overview: YOLOv8, released by Glenn Jocher in January 2023 [25], maintains high accuracy while offering faster inference speeds and introduces developer-friendly features like an easy-to-use command-line interface (CLI) [26].

b) Architecture: YOLOv8 is built on a deep convolutional neural network (CNN) architecture, consisting of a backbone (CSPNet) for feature extraction and a head for prediction. It also features an anchor-free design, which reduces box prediction numbers and improves Non-Maximum Suppression (NMS) speed [27].

6) YOLOv9:

a) Overview: YOLOv9 is the latest iteration of the YOLO (You Only Look Once) real-time object detection system [28], building on the legacy of previous versions. It incorporates advanced deep learning techniques to achieve superior accuracy, speed, and efficiency in object detection tasks.

b) Architecture: YOLOv9 introduces a novel architectural design by integrating the Generalized ELAN (GELAN) architecture with Programmable Gradient Information (PGI). This combination enhances the model's performance, making it faster and more accurate for real-time object detection.

C. Software and Libraries

1) Programming language: Python was employed for model implementation due to its extensive support for machine learning and data analysis libraries [29].

2) Deep learning frameworks:

a) PyTorch: Used for training detection models. PyTorch was chosen for its dynamic computation graph and user-friendly interface [30].

b) Tensorflow: Used for training classification models. And it was selected due to its simplicity and popularity [31].

3) Development environment and platforms:

a) Google Colab: Leveraged for its cloud-based environment, which offers accelerated computation and ease of use without the need for local installations. Google Colab was instrumental in efficiently training Python-based models [32].

b) Kaggle: Datasets were obtained from Kaggle, a leading platform for data science and machine learning competitions, offering a variety of datasets, code, and resources crucial for training, and evaluating models [33].

c) Mapillary: It is a collection of street-level pictures that has been annotated into 66 object categories and includes 25000 high-resolution images from all over the world [18].

d) Roboflow: This platform was employed for object detection and classification tasks. Roboflow assists in developing computer vision models and provides functionalities for data augmentation and model training [34].

4) Libraries and tools:

a) NumPy: Utilized for numerical computations and handling data arrays.

b) OpenCV: Used for image processing tasks, such as resizing and augmenting images, that helps in developing the field of computer vision (CV) [35].

c) CV2: It is used for image and video processing tasks, such as reading and writing images, applying filters, detecting edges, and performing image transformations.

d) Torch: Widely used for tasks like computer vision, natural language processing, and reinforcement learning.

e) Torchvision: Provide tools specifically for computer vision tasks.

f) Matplotlib: Used for creating static, interactive, and animated visualizations in Python.

D. Computational Resources

To effectively train and evaluate the object detection models, we utilized the following:

a) MacOS: M1 pro chip, CPU: 10 core GPU: 14 core, Memory: 16G

b) Windows: CPU: Intel core i7, GPU: NVIDIA GetForce MX330 2GB, Memory: 16G

E. Performance Metrics

To evaluate the performance of our models, we used the following metrics:

1) Detection models: Mean Average Precision (mAP) was used as the primary metric to assess the accuracy of object detection models. We also considered Precision and Recall.

2) Classification models: For classification models, we used Accuracy, Precision, and Recall [36].

V. RESULTS

The findings of the detection and classification models used in this research to precisely identify and detect car accidents are presented in this section. It provides a comprehensive analysis of the models' performance and effectiveness in classifying and detecting car accidents in real-world scenarios. The following subsections outline the evaluation metrics, the classification model's performance in identifying accident-related images, and the object detection model's effectiveness in localizing the involved vehicles. The results demonstrate the models' efficacy in automating the detection and classification process, contributing to more efficient accident response and improved road safety.

A. Classification

The models presented were trained using the TensorFlow deep learning framework. The categorical classes targeted by the model were "Accident" and "No-Accident". The training process utilized the Adam optimizer [37], a popular gradient-based optimization algorithm, to update the model's parameters. To evaluate the performance of our models, we employed standard evaluation metrics including accuracy, recall, and precision, and visualized these metrics using plots.

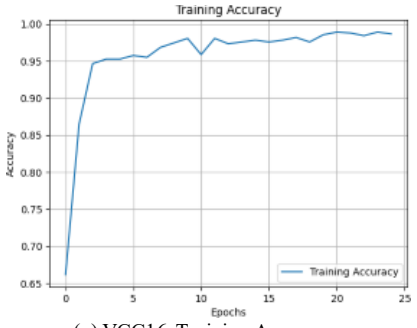
In the subsequent section, we will present the results obtained from the evaluation of classification models. The different versions of VGG16, EfficientNetB7, and ResNet101 were conducted with a consistent configuration, employing a batch size of 100, and 25 epochs for training. The proposed deep learning models were trained using TensorFlow and implemented within the Google Colab environment. As part of the training process, all training and test images were uniformly resized to dimensions of 224 x 224. Table III summarizes the results achieved in classification models with a detailed illustration of the training parameters.

TABLE III. CAR ACCIDENT CLASSIFICATION: OPTIMAL EXPERIMENTAL RESULTS

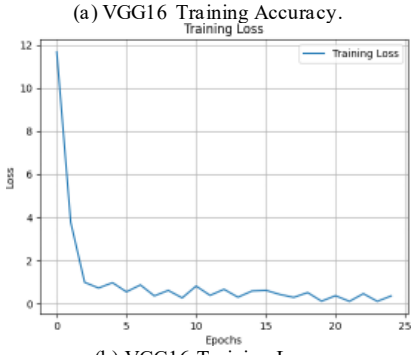
Algorithm	Epoch/ Iteration	Batch Size	Accuracy	Recall	Precision
ResNet101	25	100	93%	96%	96%
EfficientNetB7	25	100	91%	96%	96%
VGG16	25	100	92%	98%	96%

The VGG16 model, trained with 25 epochs, exhibited an impressive accuracy of 0.92. The accuracy and loss results for the training and validation are shown in Fig. 4. With a precision and a recall of 0.96 and 0.98, respectively shown in Fig. 7(a) and 8(a) depicts the confusion matrix for the VGG16 model during the training and testing phases.

Similarly, the EfficientNetB7 model achieved an accuracy of 0.91 after 25 epochs of training. The accuracy and loss trends for both training and validation iterations are illustrated in Fig. 5. With a precision and a recall of 0.96 and 0.96, respectively shown in Fig. 7(b). The confusion matrix for the EfficientNetB7 model is shown in Fig. 8(b).



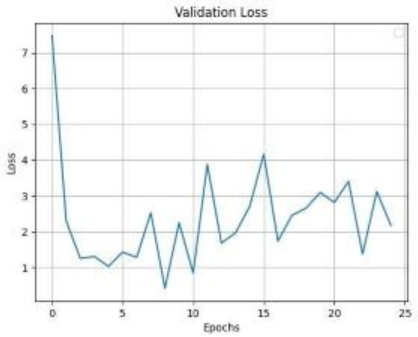
(a) VGG16 Training Accuracy.



(b) VGG16 Training Loss.



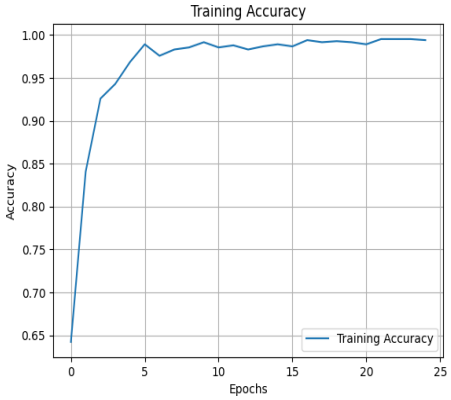
(c) VGG16 Validation Accuracy.



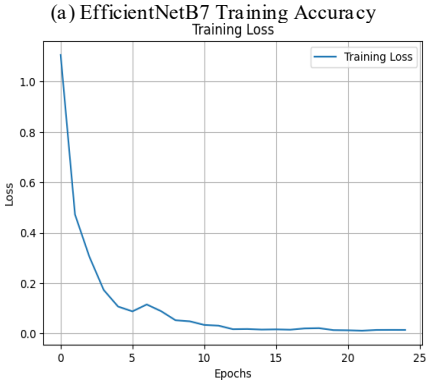
(d) VGG16 Validation Loss.

Fig. 4. Visualizing accuracy and loss of training and validation for VGG16.

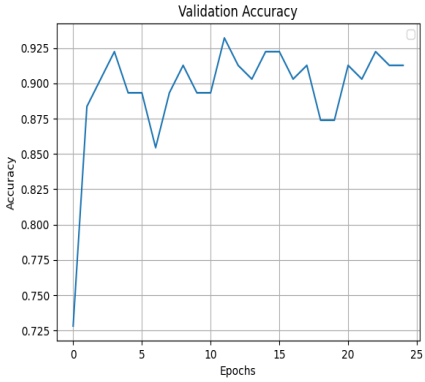
Lastly, the ResNet101 model attained accuracy of 0.93 after 25 epochs of training. The accuracy and loss curves for training and validation iterations are plotted in Fig. 6. With a precision and a recall of 0.96 and 0.96 respectively shown in Fig. 7(c). The confusion matrix for the ResNet101 model is presented in Fig. 8(c).



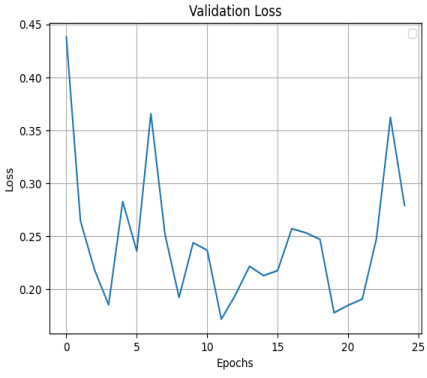
(a) EfficientNetB7 Training Accuracy



(b) EfficientNetB7 Training Loss

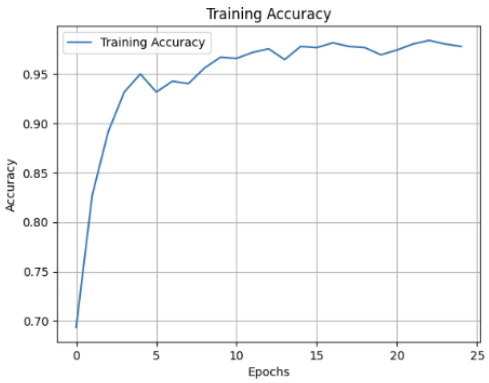


(c) EfficientNetB7 Validation Accuracy

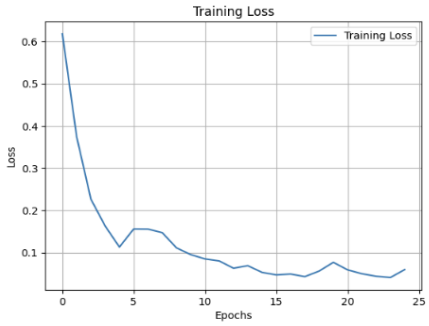


(d) EfficientNetB7 Validation Loss

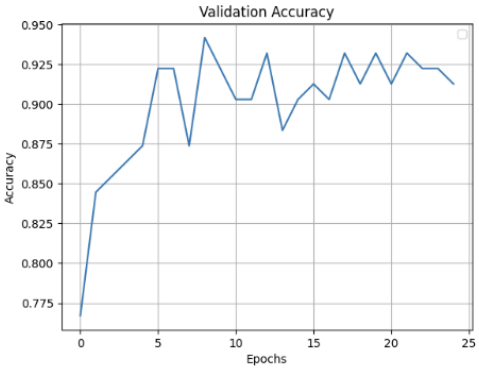
Fig. 5. Visualizing accuracy and loss of training and validation for EfficientNetB7.



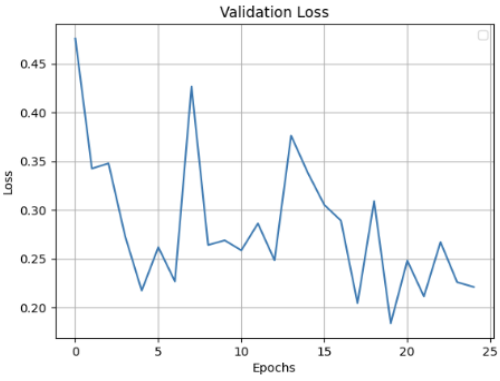
(a) ResNet101 Training Accuracy



(b) ResNet101 Training Loss

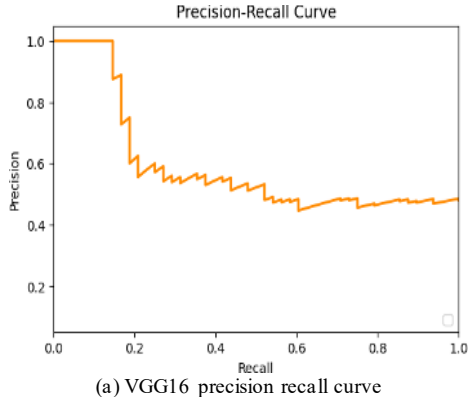


(c) ResNet101 Validation Accuracy

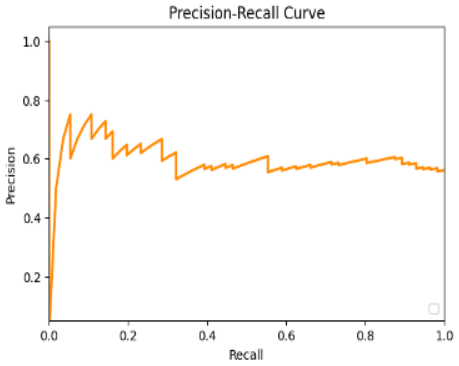


(d) ResNet101 Validation Loss

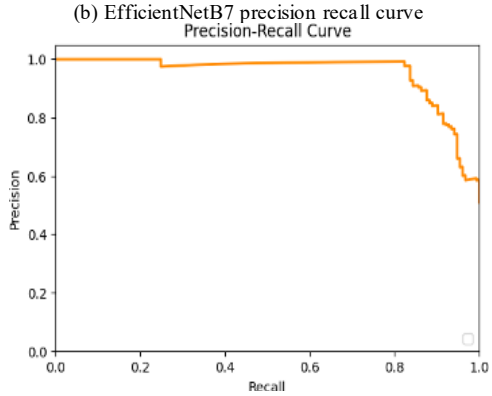
Fig. 6. Visualizing accuracy and loss of training and validation for ResNet101.



(a) VGG16 precision recall curve

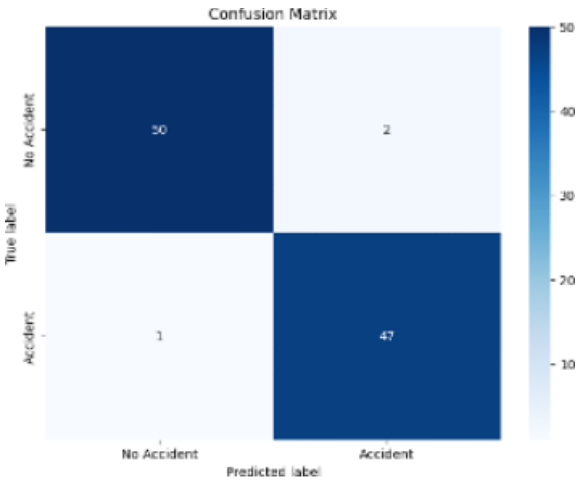


(b) EfficientNetB7 precision recall curve

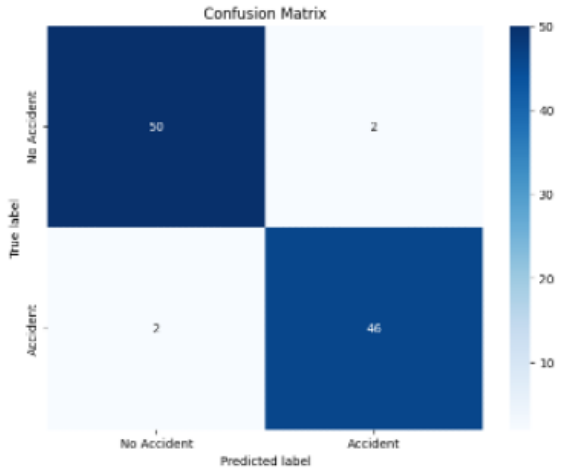


(c) ResNet101 precision recall curve

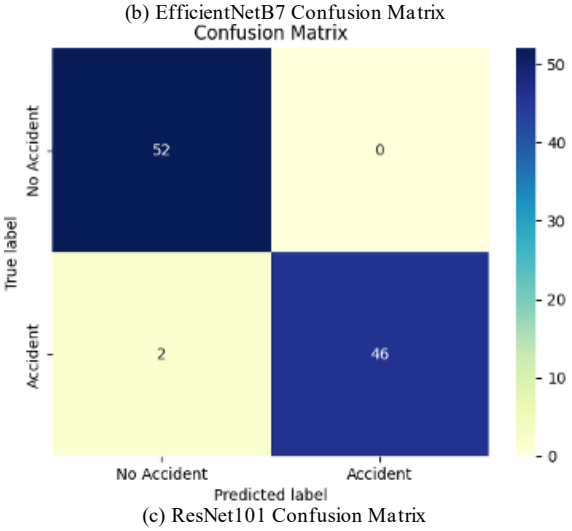
Fig. 7. Visualizing precision recall curve for VGG16, EfficientNetB7 and ResNet101.



(a) VGG16 Confusion Matrix



(b) EfficientNetB7 Confusion Matrix



(c) ResNet101 Confusion Matrix

Fig. 8. Visualizing confusion matrix for VGG16, EfficientNetB7 and ResNet101.

B. Detection

In the subsequent section, we will present the results obtained from the evaluation of detection models. The different versions of YOLO were conducted with a consistent configuration, employing a batch size of 16 and 100 epochs for training. The proposed deep learning models were trained using Pytorch and implemented within the Google Colab environment. As part of the training process, all training and test images were uniformly resized to dimensions of 640 x 640. summarization of the results achieved in detection models with a detailed illustration of the training parameters is shown in Table IV.

TABLE IV. CAR ACCIDENT DETECTION OPTIMAL EXPERIMENTAL RESULTS

Algorithm	Epoch/ Iteration	Batch Size	mAP	Recall	Precision
YOLOv5	100	16	97.8%	96.6%	95.7%
YOLOv8	100	16	95.3%	90.8%	94%
YOLOv9	100	16	95.7%	93.4%	94.1%

The YOLOv5 model, trained with 100 epochs, exhibited an impressive mean Average Precision (mAP) of 0.978. The mAP and loss results for 100 of the training and testing iterations shown in Fig. 9. With a precision and a recall of 0.957 and 0.966 respectively shown in Fig. 10(a) and (b). Fig. 11 depicts the confusion matrix for the YOLOV5 model during the training and testing phases.

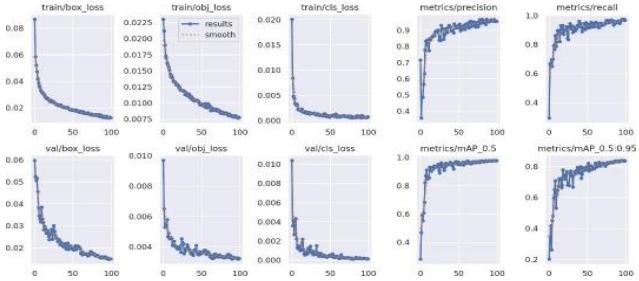
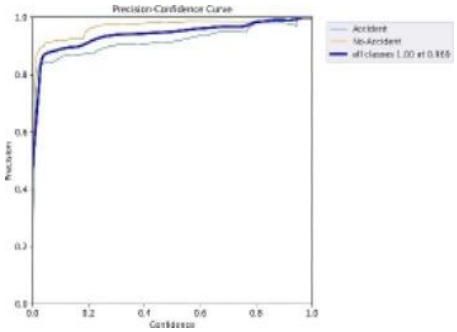
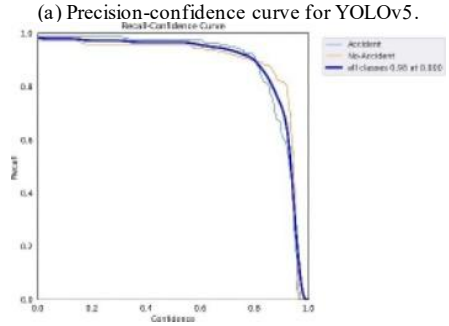


Fig. 9. mAP, training loss and validation loss for YOLOv5.



(a) Precision-confidence curve for YOLOv5.



(b) Recall-confidence curve for YOLOv5

Fig. 10. Visualizing the precision-confidence and recall-confidence curves of YOLOv5.

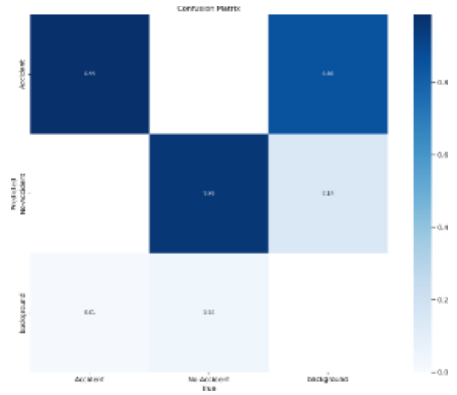


Fig. 11. Confusion matrix for YOLOv5.

The evaluation of YOLOv8 and YOLOv9 models yielded noteworthy findings regarding their mean Average Precision (mAP) values. YOLOv8 demonstrated a commendable mAP score of 0.953 in Fig. 12. Meanwhile, YOLOv9 demonstrated a notably elevated mean Average Precision (mAP) of 0.957 as illustrated in Fig. 15. Additionally, YOLOv9 showcased superior recall performance in Fig. 16(b) with a score of 0.934, whereas YOLOv8 achieved a recall value of 0.908 in Fig. 13(b). In terms of precision, YOLOv9 achieved a score of 0.941 as shown here in Fig. 16(a), while YOLOv8 attained a precision value of 0.94 in Fig. 13(a). The confusion matrices for YOLOv8 and YOLOv9 are presented in Fig. 14 and 17, respectively, providing a comprehensive visual representation of the model's performance.

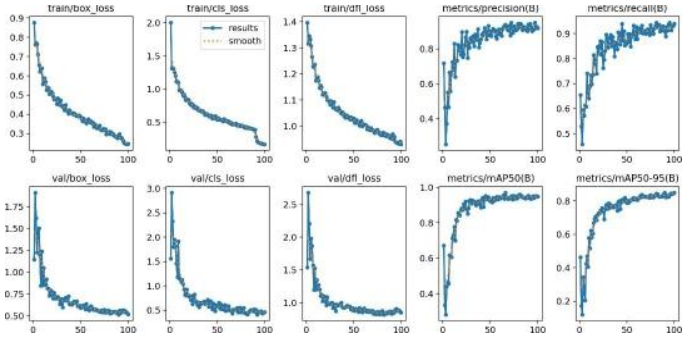
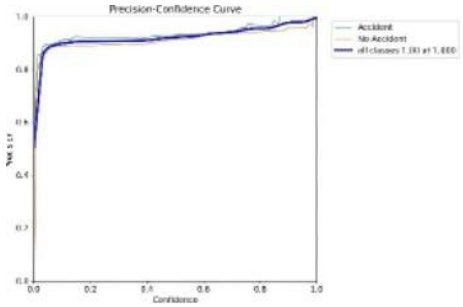
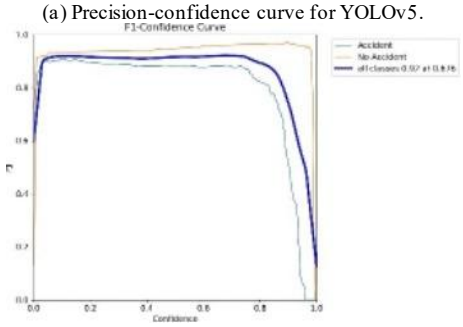


Fig. 12. mAP, training loss and Validation loss for YOLOv8.



(a) Precision-confidence curve for YOLOv5.



(b) Recall-confidence curve for YOLOv8.

Fig. 13. Visualizing the precision-confidence and recall-confidence curves of YOLOv8.

The superior performance of our YOLOv5 model compared to the newer YOLOv8 and YOLOv9 versions is a noteworthy finding. While YOLOv8 and YOLOv9 incorporate more advanced architectural features, such as the anchor-free design and the Generalized ELAN (GELAN) architecture, our results suggest that these innovations did not provide a significant advantage on our specific dataset. We hypothesize that the

simpler, well-optimized framework of YOLOv5 proved to be exceptionally effective at learning the nuanced features of our mixed dataset. This indicates that while newer models offer theoretical improvements, an earlier, well-optimized model can still achieve state-of-the-art performance, especially when its architecture is well-suited to the characteristics of the training data.

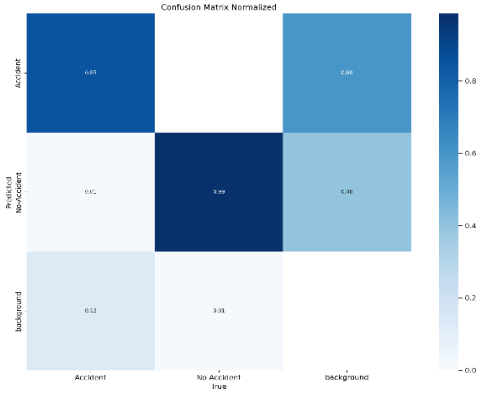


Fig. 14. Confusion matrix for YOLOv8.

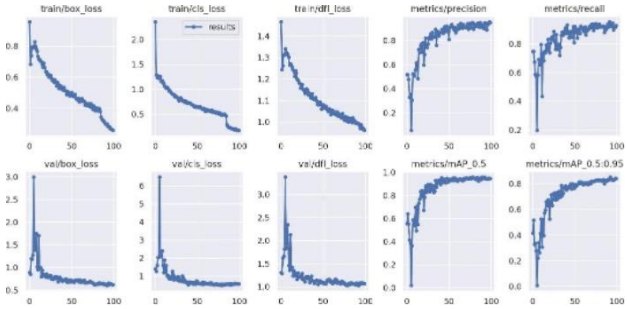
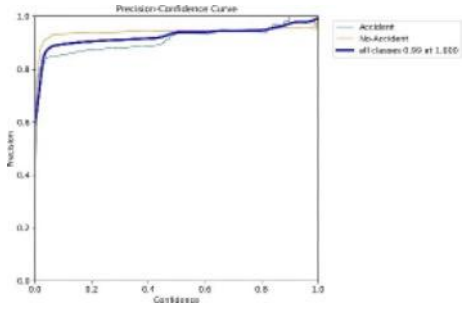
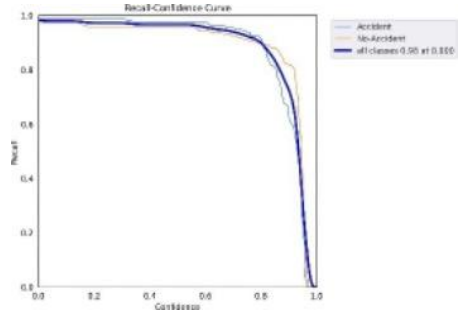


Fig. 15. mAP, training loss and validation loss for YOLOv9.



(a) Precision-confidence curve for YOLOv9.



(b) Recall-confidence curve for YOLOv9

Fig. 16. Visualizing the precision-confidence and recall-confidence curves of YOLOv9.

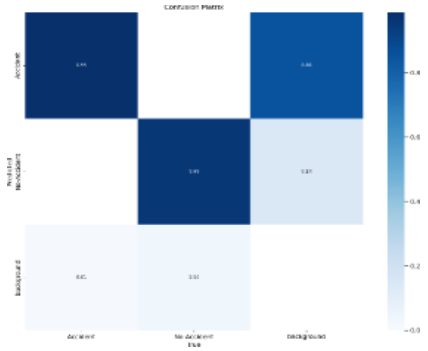


Fig. 17. Confusion matrix for YOLOv9.

VI. Discussion

The results obtained from our experiments demonstrate the effectiveness of the proposed system. A key contribution of this work is the comprehensive comparative analysis of multiple state-of-the-art deep learning models on a diverse, custom-built dataset. This approach not only identifies the most effective model for the task but also provides crucial insights into the performance trade-offs between model architecture, training time, and dataset characteristics, which is often a missing element in prior research.

According to study findings and prior research, traffic accidents rank among the top causes of death and disability worldwide. This highlights the necessity of improving urban safety through rapid accident detection and quick response. Our proposed framework contributes to solving these problems and achieving the safety of roads and communities. We trained and evaluated different deep learning models for car accident classification and detection on an extensive dataset of labeled images for "accident" and "no-accident" to assess their performance. The results obtained from our experiments demonstrate the effectiveness of the proposed system. In this section, analogous frameworks that have been conducted will be comprehensively compared.

A. Classification

In this section, we evaluate the performance of our proposed classification models:

ResNet-101, EfficientNetB7, and VGG16. The evaluation is carried out using precision, recall, and accuracy metrics which provide a comprehensive understanding of the models' classification capabilities. ResNet-101 achieved a precision and recall of 96%, with an accuracy of 93%. EfficientNetB7 exhibited a similar precision and recall of 96%, but with a slightly lower accuracy of 91%. VGG16 demonstrated a precision of 96% and a higher recall of 98%, with an accuracy of 92%. Based on these results, all three models showcase high precision values, indicating their ability to accurately classify positive instances. However, VGG16 stands out with a higher recall, suggesting its proficiency in correctly identifying positive instances while minimizing false negatives. When considering accuracy, ResNet-101 achieved the highest value of 93%, closely followed by VGG16 at 92% and EfficientNetB7 at 91%. The chart in Fig. 18 presents a comparative analysis of classification models, highlighting their performance in terms of Precision, Recall, and Accuracy metrics.

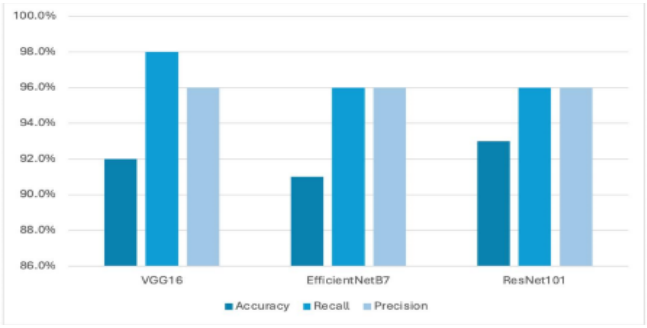


Fig. 18. Comparative analysis of classification models.

To assess the performance of our proposed model, we conducted a comparative analysis with a previously published paper [15]. The referenced paper employed the ResNet architecture for training on a dataset consisting of 500 images. They utilized ResNet-50 with a batch size of 64, a learning rate of 0.0001, and trained the model for 50 epochs. Stochastic Gradient Descent (SGD) was applied as an optimizer. The achieved accuracy was an impressive 99.3%. In our study, we trained our proposed model throughout 25 epochs. Our model attained an accuracy of 93%. This 6.3% decrease in accuracy can be attributed to several critical factors. The study's smaller, more specialized dataset likely led to a higher accuracy score due to its focused nature, while our model's slightly lower score reflects its ability to generalize to a broader range of scenarios and variations present in our diverse dataset. Additionally, the longer training duration (50 epochs) in the cited study may have allowed the model to fine-tune its parameters more thoroughly. Our comprehensive evaluation highlights the trade-offs between dataset size, diversity, training time, and overall model performance, underscoring the value of our approach on a more robust dataset.

In [11], the performance of EfficientNetB1 was evaluated using a dataset comprising 164 images. The model was trained for 300 epochs, employing a batch size of 32. The results of the study revealed that the EfficientNetB1 model achieved a commendable mAP of 89%. Contrastingly, our proposed EfficientNetB7 model surpassed the mAP reported in the study by achieving an accuracy of 91%, indicating an improvement of 2%. This is a significant finding given that our model was trained with a much shorter duration of only 25 epochs. This indicates that our model, leveraging a more advanced architecture and a larger, more diverse dataset, demonstrates enhanced performance and superior efficiency, making it a more viable solution for real-world applications where training time is a critical factor.

In another study [14], the VGG16 model was trained using a dataset consisting of 1150 images depicting damaged cars. The accuracy achieved in damage detection task was reported to be 94.56%. Notably, the study also focused on damage localization, where the VGG16 model achieved an accuracy of 74.39%. In comparison, our proposed system demonstrated encouraging results, achieving an accuracy of 92% in accident detection. This indicates that our model performs at a similar level of effectiveness as the VGG16 model utilized in the cited study, albeit with a larger dataset. The differences in dataset composition, quality, and diversity could impact the model's performance. It is possible that the dataset used in the study [14] provided a more targeted representation of damaged car images, leading to higher accuracy. Our model's ability to achieve high accuracy on a larger, more diverse dataset highlights its robust performance and adaptability to a wider range of accident scenarios. Table V summarizes the comparison between our research and other studies.

TABLE V. COMPARISON TABLE OF CLASSIFICATION MODELS

Model	Accuracy/mAP	Epochs	Batch size	Dataset
ResNet-50 [15]	99.3%	50	64	500
ResNet-101	93%	25	100	1568
EfficientNetB1 [11]	89%	300	32	164
EfficientNetB7	91%	25	100	1568
VGG16 [14]	94.56%	-	-	1150
VGG16	92%	25	100	1568

B. Detection

The evaluation conclusions show that YOLOv5 exceeds both YOLOv8 and YOLOv9 in terms of mAP, recall, and precision. It achieved the highest mAP score of 97.8%, demonstrating that it can accurately detect objects. YOLOv8 and YOLOv9 achieved slightly lower mAP scores of 95.3% and 95.7%, respectively. In terms of recall, YOLOv5 scored 96.6%, outperforming YOLOv8 and YOLOv9. This indicates that YOLOv5 performs more effectively at capturing a greater proportion of relevant objects. Similarly, YOLOv5 achieved an outstanding precision score of 95.7%, which indicates its ability to reliably detect objects with a high level of confidence. YOLOv9 had a precision score of 94.1%, which was somewhat lower than YOLOv5, while YOLOv8 had a precision score of 94%. The models were trained on the same training setup including dataset size, epochs, batch size, and image size. we can make a more direct comparison of the training times. YOLOv5 took approximately 0.408 hours to train, YOLOv8 required around 0.924 hours, and YOLOv9 took approximately 2.027 hours

Overall, YOLOv5 achieved the highest mAP, recall, and precision scores among the three models while having the shortest training duration. YOLOv8, despite slightly lower mAP, recall, and precision scores compared to YOLOv5, requires longer training time, suggesting it may not significantly outperform YOLOv5 in performance. YOLOv9, on the other hand, achieved competitive performance metrics, particularly in recall and precision, but its long training time may be a concern in time-sensitive applications or resource-constrained environments. The chart in Fig. 19 presents a comparative analysis of detection models, highlighting their performance in terms of Precision, Recall, and mAP metrics.

According to YOLOv5, we performed a comparison study with prior research [12]. The cited research utilized the YOLOv5 model trained on 8000 images of vehicles extracted from CCTV footage to detect and track vehicles. The model was trained in 900 epochs, with a batch size of 16 achieving 99.2% mAP. Additionally, the YOLOv5 was employed to detect accident severity, and it achieved 83.3% mAP. In contrast, our

YOLOv5 model was trained on 1568 annotated images of car accidents collected from several sources. Our model attained 97.8% mAP by training it with 100 epochs. These comparative results highlight the superior performance of the YOLOv5 model. This is a significant finding that highlights the importance of a carefully curated dataset. Despite using less data and a shorter training time, our model's strong performance demonstrates that the quality and diversity of our dataset—which includes a wide range of real-world and simulated accident scenarios—is a more critical factor for achieving high-performance results than sheer data quantity. Furthermore, the efficient YOLOv5 architecture, which combines high accuracy with real-time inference speeds, proved to be particularly well-suited for our dataset and task, making it a viable choice for real-world applications.

Another study employed YOLOv8 to detect windshields as a subsystem of automated detection of drivers and passengers without seat belts [38]. The dataset includes 3289 video frame images of windshields labeled for use in the windshield detection. The model trained in 100 epochs with a batch size of 16 and achieved 95% mAP. Conversely, our YOLOv8 model was trained using 1568 images in 100 epochs with a batch size of 16, achieving an impressive 95.3% mAP in the more complex task of car accident detection. This is a significant finding that highlights the importance of a well-curated dataset. While the cited study focused on a single object within a larger system, our model's performance on a smaller, yet more diverse and challenging dataset demonstrates its robust generalizability across different detection tasks. This proves that a high-quality, representative dataset can be more effective than a larger but more specialized one, a crucial insight for real-world applications where data collection can be a major challenge. The comparison of our study with previous studies is summarized in Table VI.

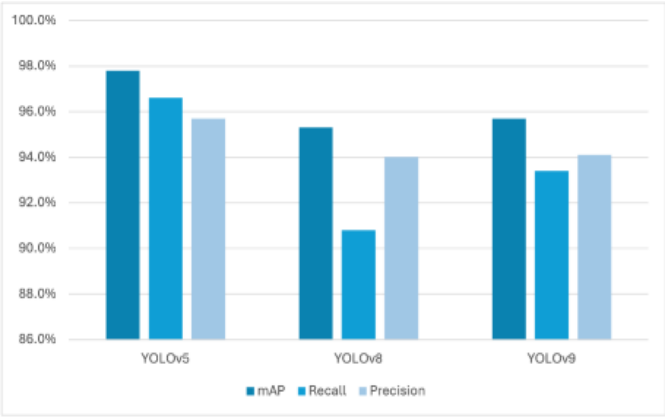


Fig. 19. Comparative analysis of detection models.

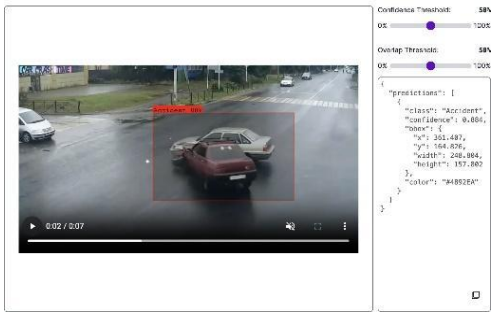
TABLE VI. COMPARISON TABLE OF DETECTION MODELS

Model	mAP	Epochs	Batch size	Dataset
YOLOv5 [12]	99.2%	900	16	8000
YOLOv5	97.8%	100	16	1568
YOLOv8 [38]	95%	100	16	3289
YOLOv8	95.3%	100	16	1568

C. Online Deployment

Considering the superior performance demonstrated by YOLOv5, a hosted video inference API developed by Roboflow is used to deploy the generated model [39]. This integration opens up a realm of possibilities for leveraging the robust capabilities of YOLOv5 in real-world applications, empowering users to harness its exceptional object detection and tracking capabilities for a wide range of video analysis tasks. The model demonstrates a remarkable ability to detect accidents within videos by meticulously analyzing individual frames and accurately assigning class labels to the objects present in the video.

Fig. 20(a) and (b) demonstrated the output of deploying the video on the hosted inference API. Each identified object is accompanied by a bounding box that precisely delineates its spatial positioning within the frames.



(a) Accident output.



(b) No-Accident output.

Fig. 20. YOLOv5 deployment on inference API.

D. Practical Implications, Limitations, and Future Direction

The findings of this research, particularly the superior performance demonstrated by YOLOv5, have significant practical implications for real-world deployment in smart cities. The model's reliability in accurately identifying and classifying car accidents is a critical first step in automating emergency response. The short training time for YOLOv5 is also a crucial factor, making it a viable solution for real-time applications and resource-constrained environments. In a smart city context, this model can be integrated with a camera and a notification system. This automated system for accident detection and classification can then swiftly notify medical facilities and security of any accidents. This would allow emergency services to react more quickly and efficiently, potentially saving lives and reducing the severity of injuries.

The findings of this research, while promising, are subject to certain limitations that warrant consideration for future work.

Firstly, the models' effectiveness may be hampered by factors such as low-quality footage or restricted camera coverage in real-world scenarios. Secondly, the computational resources required for training complex deep learning models can be a significant obstacle, as seen with our attempts to implement the Single Shot Detector (SSD) model. This limitation may affect the model's ability to reliably detect all types of accidents and can impact its effectiveness in time-sensitive applications.

The next logical step is to move from theoretical evaluation to practical implementation. This involves integrating the trained model with a camera and a notification system. Future research will also focus on expanding the dataset to include a wider variety of accident types, such as multi-vehicle collisions or accidents involving pedestrians, to improve the model's versatility. Finally, while our research focused on object detection and classification, exploring the integration of other data sources, such as real-time vehicle telemetry data, could provide a more comprehensive and accurate accident detection system in the future.

VII. Conclusions

To conclude, using different deep learning algorithms in the realm of accident detection and classification holds immense potential to enhance safety, and offers a proactive approach to accident detection, response, and prevention. In our project, we concentrated on employing diverse artificial intelligence algorithms, particularly algorithms within the realm of deep learning, to attain optimal outcomes aligned with the objectives of our project. We have employed six of the top computer vision models three of them specializing in classification task and the rest in object detection task are included to achieve our aim which is to successfully help detect and reduce accidents, thereby preserving more lives and decreasing the annual fatality rate. During the training phase, we observed satisfactory results across both classification and detection tasks. In the classification task, we obtained the following results: VGG16 model (accuracy of 0.92, precision 0.96, and recall 0.98). EfficientNetB7 model (accuracy of 0.91, precision 0.96, and recall 0.96). ResNet101 model (accuracy of 0.93, precision 0.96, and recall 0.96).

In the detection task we obtained the following results: YOLOv5 model (mAP of 0.978, precision 0.957, and recall 0.966). YOLOv8 model (mAP of 0.953, precision 0.94, and recall 0.908). YOLOv9 model (mAP 0.95, precision 0.941, and recall 0.934). According to classification results, The VGG16, EfficientNetB7, and ResNet101 models demonstrated high precision scores, with VGG16 exhibiting superior recall performance. In terms of accuracy, ResNet101 achieved the highest score among the models. And according to detection results, YOLOv5 exhibited superior performance with the highest mAP, recall, and precision scores among the three models, along with the shortest training duration.

These results underscore the efficacy of our approach in accident detection and classification tasks. The robust performance metrics, including high accuracy rates and recall scores, alongside the promising generalization capabilities of our models, highlight their potential for real-world deployment and scalability. Overall, our findings emphasize the vi ability and effectiveness of leveraging deep learning algorithms to enhance safety measures and mitigate the impact of road accidents.

Acknowledgment

The authors extend their appreciation to the Deanship of Scientific Research and Graduate Studies at Taif University, Saudi Arabia, for funding this research.

Data Availability Statement

The data that support the findings of this study are available on request from the corresponding author.

Abbreviations

The following abbreviations are used in this manuscript:

mAP - mean average precision.

SGD - Stochastic Gradient Descent.

TLA - Three letter acronym.

LD - Linear dichroism.

References

[1] MOI, "The general department of traffic (GDT) 2023." 2024. Available: https://www.moi.gov.sa/wps/portal/Home/sectors/publicsecurity/traffic/trafficriyadh/contents/!ut/p/z1/jZFPT8MwDMW_yjj0HC_JtnK0OmhCJ43ACiUXZHUbGLGMlQjYt6eHXfizrj7Z0u_Z1nvCi0r4QB_8RJG3gV7b-cGPH8FqbYZaFvki04Auu3aFXUhIx-L-JwCYXgKiujNm4uS8kML30Wc5Gj2ZAaSzfAQWTXlz7pQCVP30cKQQeuqPA757_S014urUkdZFftntPApfb0NcfUVRxYbWa64TODSDhve0fB5Qk8CGOCRAdc3LVYjv7ZO-8wToP8DUjVqguJjbqZKA8jfwT1CdQDo8AF1RnTLrbVOWZfW5V8CWLZ59A24-PZ0!/dz/d5/L0lHSkovd0RNQUprQUVnQSEhLzROVkUvZW4!/.

[2] WHO, "Reducing road crash deaths in the Kingdom of Saudi Arabia 2023." Available: https://www.who.int/news/item/20-06-2023-reducing-road-crash-deaths-in-the-kingdom-of-saudi-arabia.

[3] M. Mohammed, M. B. Khan, and E. B. M. Bashier, Machine Learning: Algorithms and Applications. Boca Raton: CRC Press, 2016.

[4] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," IEEE Trans. Intell. Transp. Syst., vol. 21, no. 11, pp. 4883–4894, November 2020.

[5] H. Luo, Q. Kong, and F. Wu, "Traffic sign image synthesis with generative adversarial networks," in 2018 24th Int. Conf. Pattern Recognit. (ICPR), Beijing, China, IEEE, 2018, pp. 2540–2545.

[6] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI '17)*, San Francisco, CA, USA, Feb. 4–9, 2017, pp. 1655–1661

[7] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Current map-matching algorithms for transport applications: state-of-the art and future research directions," Transp. Res. C Emerg. Technol., vol. 15, no. 5, pp. 312–328, October 2007.

[8] Y. Hou, P. Edara, and C. Sun, "Traffic flow forecasting for urban work zones," IEEE Trans. Intell. Transp. Syst., vol. 16, no. 4, pp. 1761–1770, August 2015.

[9] D. Q. Tran and S. H. Bae, "Proximal policy optimization through a deep reinforcement learning framework for multiple autonomous vehicles at a non-signalized intersection," Appl. Sci., vol. 10, no. 16, p. 5722, August 2020.

[10] A. O. Philip and R. K. Saravanaguru, "Multisource traffic incident reporting and evidence management in Internet of vehicles using machine learning and blockchain," Eng. Appl. Artif. Intell., vol. 117, p. 105630, January 2023.

[11] T. Tamagusko, M. G. Correia, M. A. Huynh, and A. Ferreira, "Deep learning applied to road accident detection with transfer learning and synthetic images," Transp. Res. Procedia, vol. 64, pp. 90–97, January 2022.

[12] M. I. B. Ahmed, R. Zaghdoud, M. S. Ahmed, R. Sendi, S. Alsharif, J. Alabdulkarim, B. A. A. Saad, R. Alsabt, A. Rahman, and G. Krishnasamy, "A real-time computer vision based approach to detection and classification of traffic incidents," Big Data Cogn. Comput., vol. 7, no. 1, p. 22, January 2023.

[13] V. Radu, M. Nan, M. Trăscău, D. T. Iancu, A. Ş. Ghiţă, and A. M. Florea, "Car crash detection in videos," in 2021 23rd Int. Conf. Control Syst. Comput. Sci. (CSCS), Bucharest, Romania, IEEE, 2021, pp. 127–132.

[14] P. M. Kyu and K. Woraratpanya, "Car damage detection and classification," in Proc. 11th Int. Conf. Advances Inf. Technol., Bangkok, Thailand, Association for Computing Machinery, 2020, pp. 1–6.

[15] N. Pathik, R. K. Gupta, Y. Sahu, A. Sharma, M. Masud, and M. Baz, "AI enabled accident detection and alert system using IoT and deep learning for smart cities," Sustainability, vol. 14, no. 13, p. 7701, May 2022.

[16] T. N. Le, S. Ono, A. Sugimoto, and H. Kawasaki, "Attention R-CNN for accident detection," in 2020 IEEE Intell. Vehicles Symp. (IV), Las Vegas, NV, USA, IEEE, 2020, pp. 313–320.

[17] F. Zantalis, G. Koulouras, S. Karabetsos, and D. Kandris, "A review of machine learning and IoT in smart transportation," Future Internet, vol. 11, no. 4, p. 94, March 2019.

[18] Mapillary Vistas, "Mapillary vistas dataset." Available: https://www.mapillary.com/dataset/vistas.

[19] Kaggle, "Accident detection from CCTV footage." 2020. Available: https://www.kaggle.com/datasets/ckay16/accident-detection-from-cctv-footage/data.

[20] "Car crash simulator." 2022.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR), IEEE, 2016, pp. 770–778.

[23] G. Jocher, A. Stoken, J. Borovec, L. Changyu, A. Hogan, L. Diaconu, J. Poznanski, L. Yu, P. Rai, R. Ferriday, and T. Sullivan, "ultralytics/yolov5: v3. 0, Zenodo." 2020. Available: https://zenodo.org/records/3983579.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 27–30, 2016, pp. 779–788.

[25] J. Solawetz, "What is YOLOv8? The ultimate guide." Available: https://blog.roboflow.com/what-is-yolov8/.

[26] Encord, "YOLOv8 for object detection explained [Practical example]." 2023. Available: https://medium.com/cord-tech/yolov8-for-object-detection-explained-practical-example-23920f77f66a.

[27] H. Sahota, "The history of YOLO object detection models from YOLOv1 to YOLOv8." Available: https://blogs.nvidia.com/blog/category/enterprise/deep-learning/.

[28] A. Acharya, "YOLOv9: SOTA object detection model explained." Available: https://encord.com/blog/yolov9-sota-machine-learning-object-dection-model/.

[29] Python.org 2023, "Python organiziton." Available: https://www.python.org/.

[30] N. Ketkar and J. Moolayil, "Introduction to PyTorch," in Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch, N. Ketkar and J. Moolayil, Eds. Berkeley, CA: Apress, 2021, pp. 27–91.

[31] TensorFlow, "An end-to-end platform for machine learning." Available: https://www.tensorflow.org/.

[32] Colab, "Google colaboratory." Available: https://colab.research.google.com.

[33] Kaggle, "Your machine learning and data science community." Available: https://www.kaggle.com.

[34] Roboflow, "Give your software the power to see objects in images and video." Available: https://roboflow.com.

[35] OpenCV, "Open computer vision library." Available: https://opencv.org.

[36] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," Int. J. Data Min. Knowl. Manag. Process, vol. 5, no. 2, pp. 1–11, March 2015.

[37] K. D. P. Adam and J. Ba, "A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[38] A. Sugiharto and R. Kusumaningrum, "Automated detection of driver and passenger without seat belt using YOLOv8," Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 11, pp. 806–813, November 2023.

[39] Roboflow, "What is inference? Roboflow inference." Available: https://inference.roboflow.com/.

[40] A. Chahal, S. R. Addula, A. Jain, P. Gulia, N. S. Gill, and B. V. Dhandayuthapani, "Systematic analysis based on conflux of machine learning and internet of things using bibliometric analysis," J. Intell. Syst. Internet Things, vol. 13, no. 1, pp. 196–224, Jan. 2024, doi: 10.54216/JISIoT.130115.

[41] A. Alshuaibi, M. Almaayah, and A. Ali, "Machine learning for cybersecurity issues: A systematic review," J. Cyber Secur. Risk Audit., vol. 2025, no. 1, pp. 36–46, Jan. 2025.

[42] R. S. Mousa and R. Shehab, "Applying risk analysis for determining threats and countermeasures in workstation domain," J. Cyber Secur. Risk Audit., vol. 2025, no. 1, pp. 12–21, Jan. 2025, doi: 10.63180/jcsra.thestap.2025.1.2.

Prioritizing Non-Functional Requirements and Influencing Factors for API Quality Framework: An Industry Approach

Aumir Shabbir[1], Aziz Deraman[2]*, Mohamad Nor Bin Hassan[3], Kamal Uddin Sarker[4], Shahid Kamal[5]*
Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, Kuala Nerus Terengganu, 21030, Malaysia[1, 2, 3]
Department of Business and Management Studies, Gulf College, Al Mabaila - Muscat 133, Oman[1]
Department of Computer Science and MIS, Oman College of Management and Technology, Al Barka, 320, Oman[4]
Faculty of Computing & Informatics, Multimedia University, Cyberjaya, Selangor, 63100, Malaysia[5]

***Abstract*—Application Programming Interface (API) management is currently a trending research area; however, APIs require careful attention to Non-Functional Requirements (NFRs) to ensure system performance, maintainability, security, and resiliency. The software industry struggles to maintain API quality, especially NFRs, due to a focus on functional aspects in standards like the OpenAPI Specification (OAS). Similarly, standards, such as ISO/IEC 25010:2023, evaluate the quality of general software but offer limited guidance on addressing API challenges. Based on the industry perspective, this paper prioritizes the most critical quality attributes and their influencing factors for APIs, supporting the development of a Non-Functional Requirement Quality Framework for APIs (NFRQF-API). We adopted ISO/IEC 25010 as our reference standard and surveyed industry experts. Eleven NFRs are added in the survey, including nine from ISO/IEC 25010 and two additional attributes, Observability and Resiliency, identified through the literature review. A structured survey tool has been validated, pilot-tested, and distributed to 38 API practitioners, with data analyzed through IBM Statistical Package for the Social Sciences (IBM SPSS). SPSS demonstrates strong internal consistency ($\alpha > 0.7$) across items within each group. Additionally, Maintainability (4.29) and Resiliency (4.20) have been identified as core NFRs, while Interaction Capability (3.18), Flexibility (3.18), and Safety (2.93) have lower scores based on their mean calculation. The remaining six NFRs are moderately significant, highlighting their ongoing importance. These findings, based on NFR classification, establish a solid foundation for developing a Quality Framework for APIs aligned with modern Software engineering requirements. The article supports researchers and practitioners to build a strong understanding towards NFR prioritization, which is a crucial step for API quality management.**

Keywords*—Non-Functional Requirements (NFRs); Application Programming Interface (API); software development practices; API quality; Non-Functional Requirement Quality Framework for APIs (NFRQF-API); ISO/IEC 25010

I. Introduction

In the modern landscape, APIs have become the core elements of digital ecosystems, such as cloud computing, mobile platforms, and microservices architectures. APIs cover both technical and business aspects. Technically, solutions are provided by the APIs based on the problems faced by business organizations and establish a set of requirements that determine how the application interacts and supports in the exchange of data. On the other hand, from a business perspective. APIs serve the organizations in utilizing their resources more efficiently to generate value, both across various departments and in collaboration with external partners [1]. As APIs gain significant popularity in the modern software ecosystems, their quality, explicitly focusing on NFRs, is crucial for their success [2]. However, despite their importance, organizations mainly face challenges in managing API quality, particularly regarding scalability, security, maintainability, and performance [3], [4].

NFRs are referred to as constraints and quality attributes [5], however, neglecting them can result in expensive post-development rework in software applications, which is usually a significant risk in the success of any project [6], [7]. Similarly, Observability and Resiliency have emerged as vital quality attributes in managing the complexity and reliability of modern APIs and microservices. As the behaviour of an API differs from that of traditional software, Observability as a quality attribute not only specifies the quality of the API's behaviour but also the tools, logs, and monitoring systems that provide transparency into its real-time operations [8], [9]. Likewise, Resiliency supports the graceful handling of failures without any disruptions in API services [10], [11]. Recent studies have demonstrated that integrating degradation and recovery through automated requirement-driven adaptations significantly improves overall system resilience and minimizes downtime [12].

The OpenAPI Specification (OAS) is one of the popular existing RESTful APIs standards [13], but it primarily focuses on functional aspects [14], [15], and limited to address the NFRs [16], [17]. Additionally, international quality standards, such as ISO/IEC 25010:2023, outline nine quality attributes, including Functional Suitability, Performance Efficiency, Compatibility, Interaction Capability, Reliability, Security, Maintainability, Flexibility, and Safety [18] but it mainly focuses on maintaining the quality of general software, and is limited in addressing API-specific challenges [19]. Therefore, the lack of comprehensive API guidance in these standards may lead to poor API management, which can impact quality and reliability [20]. Hence, improving API quality through a comprehensive framework that considers the broader range of quality attributes, including both design-time and runtime aspects, remains a

*Corresponding author.

significant unexplored domain in software engineering [21], [22], "in press" [23].

This study identifies and prioritizes key NFRs along with their influencing factors that affect API quality, based on industry perception. The industry feedback has been collected through a structured, validated questionnaire that underwent expert reviews, pilot testing, and statistical validation, contributing a refined classification of NFRs relevant to real-world software development. The findings based on the classification of NFRs support the development of an NFR Quality Framework for APIs, which builds a strong understanding towards the quality management of APIs for both researchers and industry.

This paper is organized into the following sections: Section II presents the literature review, Section III describes the methodology, including the comprehensive NFR prioritization process, while Section IV covers the results and a brief discussion. Similarly, Section V discusses practical and theoretical implications, whereas Section VI concludes by summarizing the findings and proposing directions for future research.

II. Literature Review

A. Significance of Application Programming Interface (API)

The API standard was first developed in the 1940s as a modular software library for EDSAC [24]. The term "Application Program Interface" was first introduced in the 1960s [25] to denote the data structures and functions associated with computer graphics on remote systems [26]. In the 2000s, as the Internet became famous, the Representational State Transfer (REST) architecture was formalized with the name of a network-based API, where the REST and similar APIs have been collected and are called "Web API" [27]. The growing significance of APIs in today's dynamic digital environment has accelerated software development [28], [29]. APIs serve as intermediaries, facilitating communication between various systems and enhancing integration, which promotes efficiency, innovation, and collaboration [30].

1) API development life cycle: APIs are not just a component that a developer can set and forget [31]. Developers need to plan, create, deploy, and monitor APIs to keep them aligned with business needs and know when to retire them. In his article, Stephen J. Bigelow [32] explains the five stages of an API lifecycle: Plan, Develop, Test, Deploy and Retire, as shown in Fig. 1:

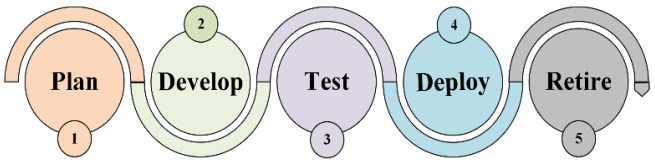


Fig. 1. API development life cycle.

B. Managing Non-Functional Requirements (NFRs)

One of the significant challenges in the dynamic world of software development is managing Non-Functional Requirements (NFRs) because of the deployment and frequent changes they face during the development lifecycle [33], [34]. NFR, as a term, was first used by Roman, G-C., in 1985 [35]. Furthermore, this term is also called quality attributes [5], goals, and non-behavioural requirements, respectively [35]. Krishnamurthy and Saran (2007) use the term "auxiliary requirements" to expand the interpretation of NFR [36]. Some NFRs are crucial and required to be addressed while designing the APIs, such as reliability, security, performance, maintainability, compatibility, and usability [37]. NFRs impact software systems' overall quality and success in a growing software development environment, where a poor approach to NFR management may be the reason for project failure [38].

Observability is one of the key attributes that support enhancing the system's reliability and performance [39]. It covers most of the post-deployment aspects of APIs, such as logging, monitoring, and metrics collection [40], [41]. Similarly, the observability also supports efficient debugging and cost management in APIs [42], [43]. Organizations can ensure that their APIs are robust, efficient, and cost-effective by following the three main components of observability, such as logging completeness, real-time monitoring, and distributed tracing [43], [44]. Similarly, the Resiliency covers fault tolerance, graceful degradation, automatic recovery, and self-healing, which are the crucial aspects in maintaining continuous service and reliability in distributed systems [45]. Fault tolerance is a crucial aspect in APIs that ensures that systems can continue their services even in the failures [12]. It provides the reliability of APIs during failures, which is essential for maintaining high availability and performance. Automatic recovery mechanisms are integral to self-healing systems, which aim to restore normal operations after a failure [46]. Kumar et al. in 2022 [47] highlight the critical importance of NFRs during software development by emphasizing that usability, security, and reliability are often overlooked or inefficiently addressed. While various existing development methodologies, such as Agile, Scrum, Spiral, DevOps, etc., mainly focus on the Functional areas [48], however, the developers ignore the crucial aspect of NFRs when designing the APIs [49], even though the quality of an application is primarily based on the NFRs [50].

Like traditional software, the quality of an API is crucial [51] that can be analyzed by evaluating multiple aspects, such as functional and non-functional [52]. The functional quality is mainly maintained by referring to the requirements that have been collected and documented. However, the Non-Functional Requirements are partially addressed, which can affect the quality of the software [53] as well as APIs [15]. Hence, software quality is essential in a software development environment and needs to be maintained frequently [54].

C. Review of Quality Factors Addressed

ISO/IEC25010 is the latest product quality model, updated in 2023 as the second edition. This model belongs to the Systems and Software Quality Requirements and Evaluation (SQuaRE) family, which applies to Information and Communication Technology (ICT) and software products. To assess the quality of the products, this model comprises nine quality characteristics [20], such as, Functional Suitability, Performance Efficiency, Compatibility, Interaction Capability, Reliability, Security, Maintainability, Flexibility, and Safety [20], as shown in Fig. 2.

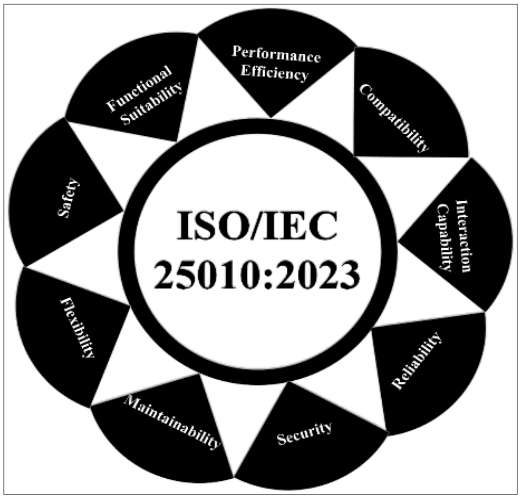


Fig. 2. A Diagrammatic model of ISO/IEC 25010:2023.

In Table I, we have analyzed few studies based on quality attributes addressed such as, Functional Suitability (FS), Performance Efficiency (PE), Compatibility (CY), Interaction Capability (IC), Reliability (RL), Security (SE), Maintainability (MY), Flexibility (FY), Safety (SA), along with the two emerging quality attributes, Observability (OY) and, Resiliency (RS).

By summarizing the review of the above-mentioned articles in terms of the comparison of quality factors addressed, Table I shows that the majority of articles only addressed Performance efficiency, Security, Maintainability, and to some extent, Functional suitability and Reliability as quality attributes in their research. However, their focus on the remaining NFRs mentioned in ISO/IEC 25010:2023 is limited. The two emerging quality attributes, Observability and Resiliency, identified in the literature review, are also addressed in a limited manner. However, the NFRQF-API effectively addresses all key non-functional quality attributes in alignment with ISO/IEC 25010.

TABLE I. LITERATURE REVIEW ON QUALITY FACTORS ADDRESSED

References	List of Quality Attributes										
	FS	***PE***	***CY***	***IC***	***RL***	***SE***	***MY***	***FY***	***SA***	***OY***	***RS***
[46]	✓	✓	□	□	✓	✓	✓	✓	□	✓	✓
[12]	✓	✓	✗	✗	✓	✓	✓	□	✗	✓	✓
[55]	✓	✓	□	✗	✓	✓	✓	□	□	✗	✗
[56]	✗	✓	✗	□	✗	✗	✗	✗	✗	✗	✗
[41]	✓	✓	✗	✗	✓	✓	✓	□	✗	✓	□
[47]	✓	✓	□	□	✓	✓	✓	□	□	✗	□
[57]	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
[58]	✗	✓	✗	□	✗	✓	✗	✗	✗	✗	✗
[59]	✗	✓	✗	□	✓	✓	✓	✗	✓	✗	✗
[60]	✗	✓	✗	□	✓	✓	✓	✓	✗	✗	✗
[61]	✗	✓	✗	□	✗	✓	✓	✗	✗	✗	✗
[62]	✗	✓	✗	□	✓	✓	✓	✗	✗	✗	✗
[63]	✗	✓	✓	□	✓	✓	✓	✗	✗	✗	✗
[64]	✓	✓	✗	✗	✓	✓	✓	✓	✗	□	□
[65]	✓	✓	□	✗	✓	✓	✓	□	✗	✗	□
[66]	✓	✓	✗	✗	✓	✓	✓	□	✗	✗	□
[67]	✓	✓	✗	✗	✓	✓	✓	□	✗	✗	□
[68]	✓	✓	□	□	✓	✓	✓	✓	✓	✗	□
[69]	✓	✓	✗	✗	✓	✓	✓	□	✗	✗	□
[70]	✓	✓	✗	✗	✓	✓	✓	□	✗	□	□
NFRQF- API	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Legend: ✓ =Fully Addressed, ✗=Not Addressed, □= Partially Addressed											

III. METHODOLOGY

This study adopts a quantitative research methodology to identify and prioritize the key NFRs that affect the quality of APIs. The quantitative approach has been selected due to its suitability in collecting expert perceptions. Fig. 3 shows a systematic structure of the Methodology adopted in this study. The structure of the Methodology consists of five phases, including Survey design, Expert Validation Process, Survey Distribution (covering both Pilot and Final Survey distribution),

Cronbach's alpha reliability test using IBM SPSS Statistics, and Data analysis (utilizing descriptive statistical tools such as mean and Standard Deviation).

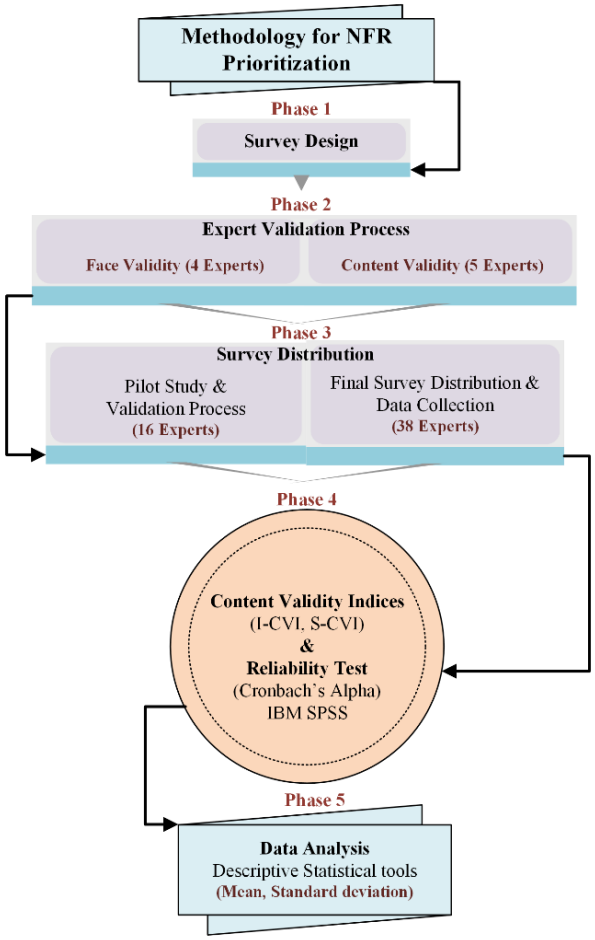


Fig. 3. A Structure of a research methodology structure.

A. Implementation of Methodology

1) Phase 1: Survey Design. The questionnaire for the pilot study comprises two primary sections: demographic data to contextualize participant responses, and targeted questions assessing the perceived importance of quality attributes mentioned in ISO/IEC 25010:2023. The first section includes 12 closed-ended and open-ended questions to collect detailed demographic data, focusing on participants' professional roles, industry sectors, and experience with API management. The second section includes 47 closed-ended questions linked to 11 quality attributes and the factors influencing them. Each quality attribute grid includes a mandatory 5-point Likert scale to assess respondent feedback, ensuring accurate and consistent responses.

2) Phase 2: Expert Validation Process. To ensure the accuracy, clarity, and appropriateness of the questionnaire, both face validity and content validity [71] assessments have been conducted prior to distribution, as mentioned below:

a) Face validity. The face validity [71], [72] of the questionnaire has been assessed through direct consultations with four academic experts possessing relevant backgrounds in research methodology, software engineering, and computer science. Their feedback has been pursued to ensure that the questionnaire is not only visually appealing and structurally sound but also suitable for effective data collection and analysis. Their feedback addressed:

- The overall clarity and understandability of the questions.
- The formatting of Likert scale responses for consistency and ease of use.
- Vocabulary adjustments to simplify technical jargon for broader respondent comprehension.
- Structural suggestions include expanding the number of questions per variable to enhance statistical reliability.

b) Content validity. Content validity [71], [73] is assessed by five domain experts in API development and software architecture, who evaluated the questionnaire for:

- Relevance and representativeness of the selected NFRs.
- Alignment with real-world industry practices, such as development environments, tools, and common development challenges.
- Addition of two novel NFRs and improvements to demographic and technical sections.

Overall, other industry experts have provided positive and encouraging feedback on the questionnaire. They also recognized that the inclusion of two emerging NFRs from the literature, along with attributes mentioned in the ISO/IEC 25010 standard, enhances the questionnaire's comprehensiveness and practicality. This feedback shows the applicability and usefulness of the questionnaire for both academic and professional domains.

3) Phase 3: Survey distribution

a) Pilot study and validation process. To ensure the instrument's quality and reliability, a pilot study [74], [75] with 16 respondents across various countries has been conducted. By following the guidelines of M. Tavakol and R. Dennick (2011) [76], a Cronbach's Alpha has been employed, which has supported identifying issues related to low reliability, such as redundancy, ambiguity, and imbalance among items. Based on the pilot results presented in Table II, overlapping or unclear questions have been refined, and the questionnaire structure has been adjusted to assign an equal number of 4 items, resulting in a total number of 44 questions. This adjustment helps to avoid potential biases in the calculation of mean scores.

b) Final survey distribution. The questionnaire-based survey methodology employed a non-probabilistic convenience sampling approach, following the expert recommendations of Wohlin, Fowler Jr, and Kasunic [77], [78], [79] to gather data from industry experts specialized in API domains across multiple countries. Convenience sampling involves selecting participants based on their accessibility and availability, with a particular focus on their expertise and relevance to API management.

c) Data collection. Data collection has occurred through an online survey platform (Google Forms), chosen for its accessibility and effectiveness in managing responses. The survey has been accessible to respondents for over four weeks, allowing industry experts sufficient flexibility to participate and complete the questionnaire conveniently. A total of 38 relevant and valid responses have been collected during the four-week period.

4) Phase 4: Content Validity Indices (I-CVI and S-CVI) and Reliability Test

a) Item-level content validity index and scale-level content validity index. Table III provides a detailed summary of the I-CVI (Item-level Content Validity Index) values for individual items and the overall S-CVI (Scale-level Content Validity Index) score, reflecting the content validity of the questionnaire.

TABLE II. GROUP-WISE CRONBACH'S ALPHA RELIABILITY FOR NFR ITEMS (PILOT STUDY)

Sr.#	NFRs	Number of Items	Cronbach's Alpha (Pilot Study)
1.	FS	5	0.61
2.	PE	4	0.57
3.	CY	4	0.41
4.	IC	5	0.72
5.	RL	3	0.48
6.	SE	4	0.75
7.	MY	5	0.54
8.	FY	3	0.52
9.	SA	5	0.50
10.	OY	4	0.80
11.	RS	5	0.77

TABLE III. I-CVI (ITEM-LEVEL CVI) AND S-CVI (SCALE-LEVEL CVI)

NFR	Factors	Experts rating 3 or 4	I-CVI
FS	FS1: It is important that the API provides all the required functions for users.	5	1.00
	FS2: It is important that the API covers all tasks and goals thoroughly.	4	0.80
	FS3: It is important that the API provides accurate and reliable results.	5	1.00
	FS4: It is essential that Functional Suitability supports API quality by allowing tasks to be completed without unnecessary steps.	5	1.00
PE	PE1: It is important that the API provides a quick response time under expected load conditions.	4	0.80
	PE2: It is important that the API ensures minimal resource usage during operations.	5	1.00
	PE3: It is important that the API efficiently handles high traffic, ensuring performance under peak conditions.	5	1.00
	PE4: It is essential that Performance Efficiency enhances API speed, responsiveness, and resources used to ensure quality.	4	0.80
CY	CY1: It is important that the API integrates seamlessly with other systems, regardless of platform.	4	0.80
	CY2: It is important that the API supports multiple platforms (e.g., mobile, web, IoT) and environments.	5	1.00
	CY3: It is important that the API ensures backward compatibility with previous versions.	4	0.80
	CY4: It is essential that Compatibility supports API quality by ensuring seamless integration with multiple platforms and systems.	5	1.00
IC	IC1: It is important that the API's interface is intuitive and easy for users to operate and control.	4	0.80
	IC2: It is important that the API provides meaningful error messages for troubleshooting.	4	0.80
	IC3: It is important that users can quickly learn to use the API.	4	0.80
	IC4: It is essential that an API's quality depends on strong Interaction Capability, including clear documentation, a user-friendly interface, and easy integration.	4	0.80
RL	RL1: It is important that the API consistently performs its intended functions without failure, even under load.	5	1.00
	RL2: It is important that the API ensures high availability with minimal downtime to maintain service continuity.	5	1.00
	RL3: It is important that the API remains operational even in the event of an error.	5	1.00
	RL4: It is essential that Reliability supports API quality by ensuring consistent performance, high availability, and minimal downtime.	4	0.80
SE	SE1: It is important that the API provides strong authentication and authorization to secure access.	5	1.00
	SE2: It is important that the API uses data encryption during transmission and storage to protect sensitive information.	5	1.00
	SE3: It is important that the API is resilient against common security threats, such as injection and cross-site scripting attacks.	4	0.80
	SE4: It is essential that Security supports API quality by protecting data, preventing unauthorized access, and meeting security standards.	4	0.80
MY	MY1: It is important that the API has a clear modular architecture to support easy maintenance and upgrades.	5	1.00
	MY2: It is important that the API allows seamless updates and versioning without disrupting service or breaking backward compatibility.	5	1.00
	MY3: It is important that well-structured and clear documentation helps maintain and troubleshoot the API efficiently.	5	1.00

	MY4: It is essential that Maintainability enables easy updates, bug fixes, and adaptation to new requirements over time.	5	1.00
FY	FY1: It is important that the API supports multiple deployment models (e.g., cloud, on-premises).	4	0.80
	FY2: It is important that the API can be installed easily.	4	0.80
	FY3: It is important to design APIs that are scalable to support flexibility.	4	0.80
	FY4: It is essential that the API stays flexible so it can adapt to new business needs and changing technologies.	5	1.00
SA	SA1: It is important that the API does not cause harm to people, property, or the environment.	4	0.80
	SA2: It is important that the API works within safe limits to prevent unsafe actions.	4	0.80
	SA3: It is important that the API can help find risks that could cause harm.	4	0.80
	SA4: It is essential that Safety helps ensure API quality by preventing unsafe actions and switching to a safe mode if something goes wrong.	4	0.80
OY	OY1: It is important that the API captures and retains logs for debugging, auditing, and monitoring performance.	5	1.00
	OY2: It is important that the API provides real-time monitoring and alerts to notify of any failures.	4	0.80
	OY3: It is important that the API uses distributed tracing to track requests across services.	5	1.00
	OY4: It is essential that Observability supports API quality by enabling monitoring, logging, and troubleshooting to maintain optimal performance.	5	1.00
RS	RS1: It is important that the API handles failures smoothly without disrupting service.	5	1.00
	RS2: It is important that the API keeps partial functionality available instead of failing completely.	5	1.00
	RS3: It is important that the API can recover automatically from failures without requiring manual intervention.	5	1.00
	RS4: It is essential that Resiliency supports API quality by effectively managing failures, recovering quickly, and maintaining uninterrupted operations.	5	1.00
S-CVI (Average of I-CVIs)			**0.91**

N= Number of Experts = 5

M= Number of Items = 44

A_i= Number of experts who rated item (i) as 3 or 4

$$I-CVI_i = \frac{Number\ of\ experts\ rating\ 3\ or\ 4\ (A_i)}{Total\ number\ of\ Experts}$$

IF, (A_i= 5 and N= 5) $I-CVI_i\ (for\ 24\ items) = \frac{5}{5},\ I-CVI_i = 1.00$

IF, (A_i= 4 and N= 5)

$$I-CVI_i\ (for\ 20\ items) = \frac{4}{5}, \qquad I-CVI_i = 0.80$$

$$S-CVI_{Ave} = \frac{\sum (I-CVI\ for\ all\ items)}{Total\ number\ of\ Items\ (M)}$$

Hence,

$$S-CVI_{Ave} = \frac{40}{44}, \qquad S-CVI_{Ave} = 0.91$$

To ensure the relevance and appropriateness of the questionnaire items before the final distribution, a Content Validity Index (CVI) has been calculated based on ratings from five domain experts. Each expert has rated all 44 items on a 4-point scale (1 = Not Relevant to 4 = Highly Relevant). An item-level CVI(I-CVI) has been computed by dividing the number of experts who rated an item as 3 or 4 by the total number of experts. All 44 items have achieved an I-CVI ≥ 0.78, indicating acceptable content validity based on the standard threshold recommended by Lynn (1986) [80] and supported by the recent studies of Almohanna et al., (2022) [81]. Thus, no items have been eliminated at this stage. By following the guidelines of Polit, D. F., & Beck, C. T. (2006), the Scale-level Content Validity Index (S-CVI) has been calculated as the average of all I-CVI values across the instrument, which shows 0.91, indicating excellent overall content validity [73].

b) Reliability test (Cronbach's alpha). By following the guidelines of M. Tavakol and R. Dennick (2011) [76], a Cronbach's Alpha is employed to assess the reliability of the questionnaire shown in Table IV, followed by Fig. 4, which ensures that the set of questionnaire items is reliable.

The above-mentioned Table IV and Fig. 4 demonstrate the acceptable reliability ($\alpha > 0.7$) [76], [82] of the items mentioned against each NFR. Reliability coefficients across all NFRs in the final study exceeded the acceptable threshold of 0.70, with Compatibility (CY), Security (SE), and Observability (OY) achieving the highest reliability, above 0.80. However, Functional Suitability (FS), Performance Efficiency (PE), Interaction Capability (IC), Reliability (RL), Maintainability (MY), Flexibility (FY), Safety (SA), and Resiliency (RS) remained between 0.70 and 0.80, indicating good reliability coefficients.

5) Phase 5: Data Analysis. The collected data have been prepared for analysis through coding and categorization based on participant responses to Likert-scale items. By applying IBM SPSS Statistics, descriptive statistical tools such as mean and standard deviation are used to compute the ranking of NFRs for identification purposes. Table V, together with Fig. 5, below, shows the mean and standard deviation range to compute the ranking of NFRs. By following the guidelines of "Boone & Boone, 2012" [83], an interpretation based on the mean and standard deviation has also been included in the table.

TABLE IV. GROUP-WISE CRONBACH'S ALPHA RELIABILITY FOR NFR ITEMS (FINAL STUDY)

Sr.#	NFRs	Final Study	
		Number of Items	*Cronbach's Alpha*
1.	FS	4	0.72
2.	PE	4	0.74
3.	CY	4	0.81
4.	IC	4	0.73
5.	RL	4	0.76
6.	SE	4	0.82
7.	MY	4	0.75
8.	FY	4	0.76
9.	SA	4	0.76
10.	OY	4	0.82
11.	RS	4	0.79

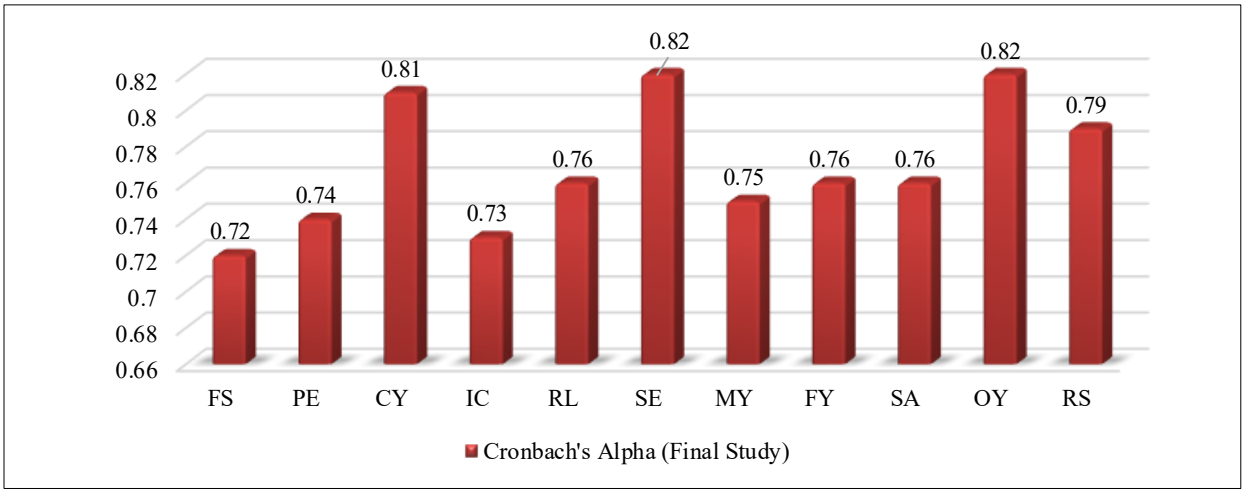


Fig. 4. Group-wise Cronbach's alpha reliability for NFR items (final study).

TABLE V. OVERALL MEAN AND STANDARD DEVIATION FOR 11 NFRS

NFR	Mean	Standard Deviation	Interpretation
MY	4.29	0.58	High Importance, High Consensus
RS	4.20	0.69	High Importance, High Consensus
FS	4.18	0.54	Moderate Importance, High Consensus
OY	4.16	0.61	Moderate Importance, High Consensus
SE	4.14	0.65	Moderate Importance, High Consensus
RL	4.09	0.62	Moderate Importance, High Consensus
PE	4.08	0.65	Moderate Importance, High Consensus
CY	3.97	0.66	Moderate Importance, High Consensus
IC	3.18	0.68	Low to Moderate Importance, High Consensus
FY	3.18	0.55	Low to Moderate Importance, High Consensus
SA	2.93	0.63	Low to Moderate Importance, High Consensus

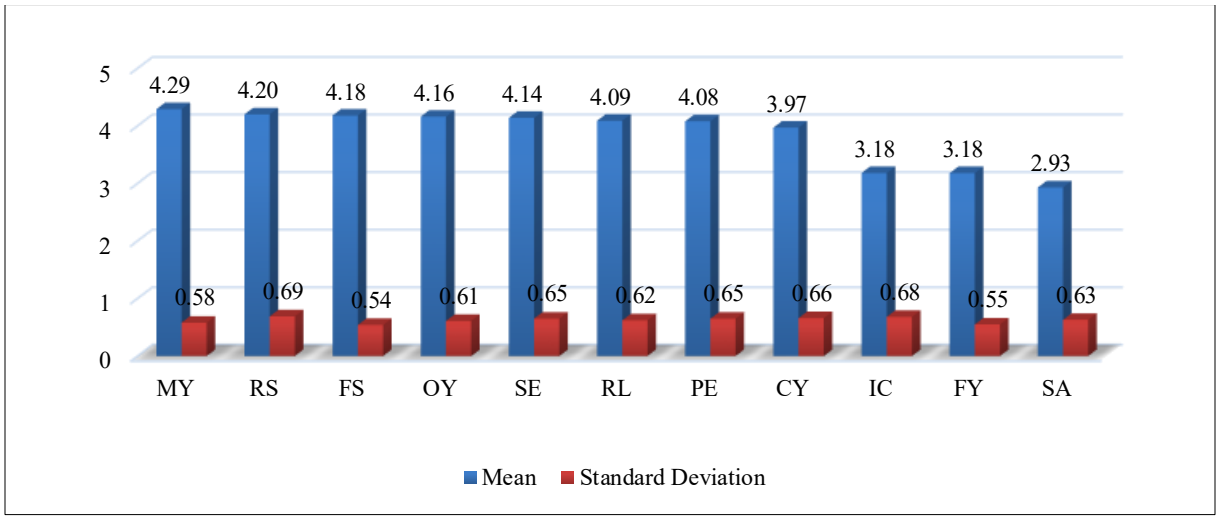


Fig. 5. NFR-wise mean and standard deviation.

The above-mentioned Table V, followed by Fig. 5 shows that Maintainability and Resiliency achieve "High Importance" as Core NFRs, while Functional Suitability (FS), Observability (OY), Security (SE), Reliability (RL), Performance Efficiency (PE), and Compatibility (CY) are considered to have moderate importance and are regarded as Critical NFRs. Conversely, Interaction Capability (IC), Flexibility (FY), and Safety (SA) are assessed as "Low to Moderate Importance" and are interpreted as Contextual NFRs. On the other hand, the standard deviation indicates a high level of variability in participants' responses to each item in the questionnaire.

For a detailed analysis of the factors influencing each NFR, Table VI offers an interpretation of each factor based on the mean and standard deviation.

TABLE VI. INTERPRETATION BASED ON FACTOR-WISE MEAN AND STANDARD DEVIATION

NFR	Factors	Mean	Standard Deviation	Mean Interpretation	Standard Deviation Interpretation
FS	FS1	4.29	0.65	High Importance	High Consensus
	FS2	4.03	0.82	Moderate Importance	Moderate Consensus
	FS3	4.03	0.79	Moderate Importance	High Consensus
	FS4	4.37	0.63	High Importance	High Consensus
PE	PE1	4.18	0.83	Moderate Importance	Moderate Consensus
	PE2	3.87	0.96	Moderate Importance	Moderate Consensus
	PE3	4.03	1.00	Moderate Importance	Moderate Consensus
	PE4	4.24	0.63	High Importance	High Consensus
CY	CY1	3.95	0.87	Moderate Importance	Moderate Consensus
	CY2	4.26	0.72	High Importance	High Consensus
	CY3	3.79	0.87	Moderate Importance	Moderate Consensus
	CY4	3.87	0.84	Moderate Importance	Moderate Consensus
IC	IC1	3.21	0.93	Low to Moderate Importance	Moderate Consensus
	IC2	3.18	0.93	Low to Moderate Importance	Moderate Consensus
	IC3	3.11	0.89	Low to Moderate Importance	Moderate Consensus
	IC4	3.24	0.88	Low to Moderate Importance	Moderate Consensus
RL	RL1	3.87	0.99	Moderate Importance	Moderate Consensus
	RL2	4.05	0.80	Moderate Importance	Moderate Consensus
	RL3	4.05	0.80	Moderate Importance	Moderate Consensus
	RL4	4.37	0.63	High Importance	High Consensus
SE	SE1	4.18	0.8	Moderate Importance	Moderate Consensus
	SE2	4.11	0.86	Moderate Importance	Moderate Consensus

	SE3	3.95	0.87	Moderate Importance	Moderate Consensus
	SE4	4.34	0.71	High Importance	High Consensus
MY	MY1	4.21	0.74	High Importance	High Consensus
	MY2	4.05	0.90	Moderate Importance	Moderate Consensus
	MY3	4.53	0.65	High Importance	High Consensus
	MY4	4.37	0.75	High Importance	High Consensus
FY	FY1	3.11	0.80	Low to Moderate Importance	Moderate Consensus
	FY2	3.16	0.79	Low to Moderate Importance	High Consensus
	FY3	3.24	0.63	Low to Moderate Importance	High Consensus
	FY4	3.21	0.66	Low to Moderate Importance	High Consensus
SA	SA1	2.84	0.82	Low to Moderate Importance	Moderate Consensus
	SA2	2.84	0.89	Low to Moderate Importance	Moderate Consensus
	SA3	3.29	0.69	Low to Moderate Importance	High Consensus
	SA4	2.74	0.86	Low to Moderate Importance	Moderate Consensus
OY	OY1	4.16	0.82	Moderate Importance	Moderate Consensus
	OY2	4.05	0.84	Moderate Importance	Moderate Consensus
	OY3	4.18	0.69	Moderate Importance	High Consensus
	OY4	4.26	0.69	High Importance	High Consensus
RS	RS1	3.97	0.91	Moderate Importance	Moderate Consensus
	RS2	4.32	0.84	High Importance	Moderate Consensus
	RS3	4.32	0.90	High Importance	Moderate Consensus
	RS4	4.21	0.87	High Importance	Moderate Consensus

The above-mentioned Table VI shows an in-depth analysis of the factors associated with each NFR, where each NFR is divided into four items based on the factors affecting it. Table VI presents the Mean, Standard deviation, and interpretation based on the mean and standard deviation of the four items for each NFR. Out of the 11 NFRs (44 items), 8 NFRs (32 items) are categorized as having “High” or “Moderate” importance. However, 3 NFRs (12 items) specifically Interaction Capability (IC), Flexibility (FY), and Safety (SA) are assessed as having "Low to Moderate" importance.

IV. Results and Discussion

A. Comparison of Pilot and Final Studies

Table VII, followed by Fig. 6 below, summarizes the key differences between pilot and final reliability metrics.

TABLE VII. Group-Wise Cronbach’s Alpha Reliability Comparison of Pilot and Final Study

Sr.#	NFR	Pilot Study		Final Study	
		Number of Items	*Cronbach's Alpha*	*Number of Items*	*Cronbach's Alpha*
1.	FS	5	0.61	4	0.72
2.	PE	4	0.57	4	0.74
3.	CY	4	0.41	4	0.81
4.	IC	5	0.72	4	0.73
5.	RL	3	0.48	4	0.76
6.	SE	4	0.75	4	0.82
7.	MY	5	0.54	4	0.75
8.	FY	3	0.52	4	0.76
9.	SA	5	0.50	4	0.76
10.	OY	4	0.80	4	0.82
11.	RS	5	0.77	4	0.79

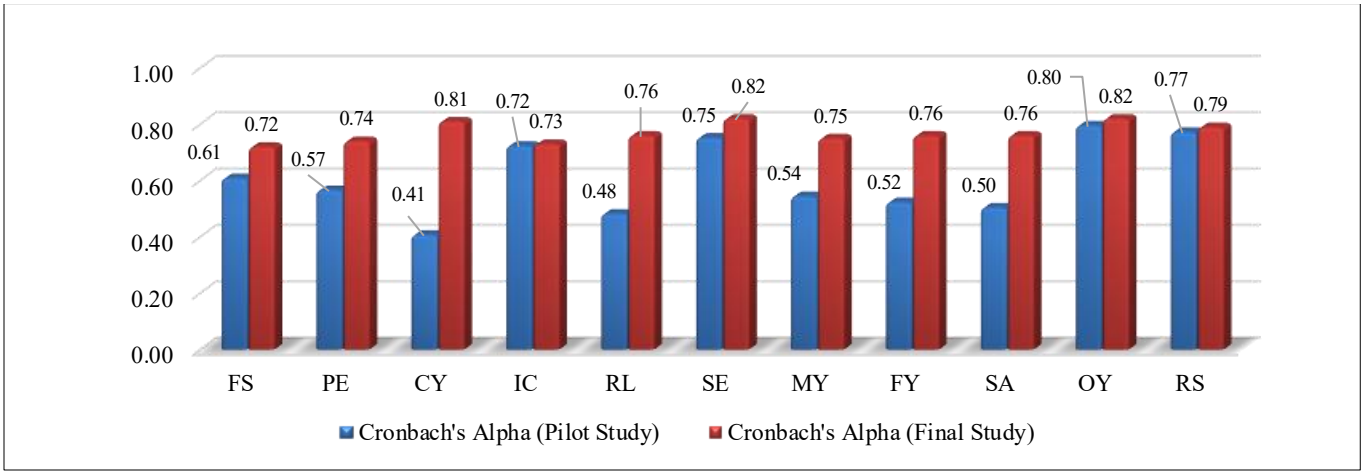


Fig. 6. Group-wise Cronbach's alpha reliability comparison of pilot and final study.

Based on the comparison between the Pilot and Final studies mentioned in the above Table VII, the final survey achieved higher Cronbach's alpha scores and confirmed improved internal consistency, with group-wise reliability for each NFR exceeding the threshold of 0.70, demonstrating strong reliability across all individual dimensions.

The Item-Level Content Validity Index (I-CVI) was calculated as the proportion of experts rating each item as three or four. Twenty-four items were rated three or four by all five experts, resulting in an I-CVI of 1.00. Conversely, 20 items received ratings of three or four from four experts, resulting in an I-CVI of 0.80. Consequently, all items exceeded the recommended minimum of 0.78 for five reviewers. Using the average method, the Scale-Level Content Validity Index (S-CVI_{Ave}) also confirmed the excellent overall content validity of the instrument as 0.91.

B. Prioritization of NFRs

Analysis of expert responses based on the Descriptive analysis of NFRs (mentioned in Tables V and VI) shows:

- Maintainability (MY) and Resiliency (RS) have been rated as highly important (Mean ≥ 4.20) with strong consensus (SD ≤ 0.79) and are classified as Core NFRs.
- Functional Suitability (FS), Performance Efficiency (PE), Compatibility (CY), Reliability (RL), Security (SE), and Observability (OY) demonstrate moderate importance (Mean 3.40 – 4.19), with varying levels of consensus and are categorized as Critical NFRs.
- Interaction Capability (IC), Flexibility (FY), and Safety (SA) are rated as low to moderate significance (Mean 2.60 – 3.39), with standard deviations reflecting considerable variability in responses and are classified as Contextual NFRs.
- The factor-wise analysis also reveals similar trends: the dimensions related to MY and RS were rated as highly important, while those associated with IC, FY, and SA scored comparatively lower.

C. Validation and Related Work

The classification of NFRs into Core, Critical, and Contextual categories was validated through experts' feedback, as demonstrated by the results of descriptive statistics. These results align with recent research that identifies Maintainability [84], [85] and Resiliency [10] as the essential NFRs of API quality and long-term adaptability [86]. Other studies highlight Resiliency as a key attribute for modern microservice and cloud-native APIs, emphasizing fault tolerance and automatic recovery to ensure uninterrupted service [11], [12]. Similar support exists for Functional Suitability, Performance Efficiency, Compatibility, Reliability and Security as primary quality attributes in API design [85], [3], [4]. Likewise, observability as a quality attribute not only defines the behaviour of the API but also covers the tools, logs, and monitoring systems that offer transparency into its real-time operations [8], [9]. By integrating these findings with the ISO/IEC 25010 quality model and expanding it to incorporate emerging factors such as Observability and Resiliency, this study bridges established standards with current industry needs, offering both theoretical foundation and practical relevance.

V. Practical and Theoretical Implications

This research offers both practical and theoretical contributions through the presentation of a validated prioritization of NFRs for API development. These requirements are categorized as Core, Critical, and Contextual, aligning with ISO/IEC 25010 while incorporating emerging NFRs such as Observability and Resiliency. The classification supports industry professionals in early planning of high- and medium-prioritized NFRs and their influencing factors during the development phases of APIs. Conversely, lower-prioritized Contextual NFRs are addressed as context-based on specific project requirements. Furthermore, theoretically, it enhances the literature by integrating emerging API-specific NFRs with an international standard, offering researchers with a structured and empirically supported foundation for future studies and for developing a comprehensive NFR-based API quality framework.

VI. Conclusion and Future Work

This study introduces a practical and validated instrument for capturing industry insights on API-related NFRs. The comprehensive pilot process, expert feedback, and strong internal consistency verify that the instrument is reliable and relevant for guiding future framework development. Moreover, our study successfully prioritize the most critical NFRs that influence API quality, as determined by relevant industry experts. We have gathered feedback from 38 respondents with considerable expertise in the API domain. The key findings emphasize that Maintainability and Resiliency are top priorities as core NFRs, reflecting the need for APIs that are easy to update, troubleshoot, and recover after failures. Meanwhile, moderate importance for critical NFRs highlights that Functional Suitability, Performance Efficiency, Compatibility, Reliability, Security, and Observability remain essential. Considering the low ratings for Interaction Capability, Flexibility, and Safety, our proposed framework should focus on developer-focused usability, in addition to the user-focused usability defined under Interaction Capability in the ISO standard. Furthermore, the framework will also assess whether the installability, as mentioned in the Flexibility and physical safety aspects of Safety, genuinely applies to the API domain.

Based on the preliminary findings, future research should focus on developing a comprehensive Non-Functional Requirement Quality Framework for APIs (NFRQF-API) that aligns with current quality standards. Additionally, these insights can significantly support both academic research and industry practices, offering a solid foundation for developing an API Non-Functional Requirements Quality Framework for APIs that aligns well with modern software engineering approaches and practical implementation needs.

Acknowledgment

We sincerely thank Universiti Malaysia Terengganu (UMT) and the Ministry of Higher Education (MOHE) Malaysia for their support in conducting this research. We also appreciate Gulf College, Muscat, Oman and the Ministry of Higher Education, Research and Innovation - Sultanate of Oman for their vital research assistance.

References

[1] S. Andreo and J. Bosch, "API Management Challenges in Ecosystems," in Eur. Commission, Louxembourg, Louxembourg, UK, Tech. Rep. JRC118082, 2019.

[2] A. Rashwan, "Automated quality assurance of non-functional requirements for testability," 2015.

[3] S. Andreo and J. Bosch, "API Management Challenges in Ecosystems," in Software Business: 10th International Conference, ICSOB 2019, Jyväskylä, Finland, November 18–20, 2019, Proceedings 10, Springer, 2019.

[4] M. Ahmad, J. J. Geewax, A. Macvean, D. Karger, and K.-L. Ma, "API Governance at Scale," in Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice, New York, NY, USA: ACM, Apr. 2024.

[5] F.-L. Li et al., "Non-functional requirements as qualities, with a spice of ontology," in 2014 IEEE 22nd International Requirements Engineering Conference (RE), IEEE, Aug. 2014.

[6] X. Zhang and X. Wang, "Tradeoff Analysis for Conflicting Software Non-Functional Requirements," IEEE Access, vol. 7, pp. 156463–156475, 2019.

[7] U. S. Shah, S. J. Patel, and D. C. Jinwala, "Constructing a Knowledge-Based Quality Attributes Relationship Matrix to Identify Conflicts in Non-Functional Requirements," J. Comput. Theor. Nanosci., vol. 17, no. 1, pp. 122–129, Jan. 2020.

[8] M. Cinque, R. Della Corte, and A. Pecchia, "Microservices Monitoring with Event Logs and Black Box Execution Tracing," IEEE Trans. Serv. Comput., vol. 15, pp. 294–307, 2019.

[9] M. Scrocca, R. Tommasini, A. Margara, E. Della Valle, and S. Sakr, "The Kaiju project: enabling event-driven observability," Proc. 14th ACM Int. Conf. Distrib. Event-based Syst., 2020.

[10] J. Zhang, Y. Rong, J. Cao, C. Rong, J. Bian, and W. Wu, "DBFT: A Byzantine Fault Tolerance Protocol With Graceful Performance Degradation," IEEE Trans. Dependable Secur. Comput., vol. 19, pp. 3387–3400, 2021.

[11] Y. Lin, S. Kulkarni, and A. Jhumka, "Automation of fault-tolerant graceful degradation," Distrib. Comput., vol. 32, pp. 1–25, 2019.

[12] P. Rajput and G. Sikka, "Multi-agent architecture for fault recovery in self-healing systems," J. Ambient Intell. Humaniz. Comput., vol. 12, pp. 2849–2866, 2020.

[13] A. Shabbir, N. Tahir, M. Shahzad, and T. D. A. Deraman, "Perceptions on Restful APIS Testing and Evaluation for Digitial AI Deployed Applications-A Literature Review and Systematic Survey," Well Test. J., vol. 33, no. S2, pp. 551–570, 2024.

[14] A. Karavisileiou, N. Mainas, and E. G. M. Petrakis, "Ontology for OpenAPI REST Services Descriptions," in 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, Nov. 2020.

[15] S. Bucaille, J. L. Cánovas Izquierdo, H. Ed-Douibi, and J. Cabot, "An OpenAPI-Based Testing Framework to Monitor Non-functional Properties of REST APIs," in International Conference on Web Engineering, Springer, 2020.

[16] R. Pergl and N. Jíša, "Semantic Analysis of API Blueprint and OpenAPI Specification," in World Conference on Information Systems and Technologies, Springer, 2024.

[17] A. Gamez-Diaz et al., "The role of limitations and SLAs in the API industry," in Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2019, pp. 1006–1014.

[18] R. Nacheva, "Conceptual Model of a Software Accessibility Evaluation System".

[19] E. dos Santos and S. Casas, "API Management and SQuaRE: A Comprehensive Overview from the Practitioners' Standpoint," in Argentine Congress of Computer Science, Springer, 2023, pp. 137–150.

[20] ISO, "ISO/IEC 25010:2023," Online Browsing Platform (OBP). [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-2:v1:en

[21] L. Murphy, M. B. Kery, O. Alliyu, A. Macvean, and B. A. Myers, "API designers in the field: Design practices and challenges for creating usable APIs," in 2018 ieee symposium on visual languages and human-centric computing (vl/hcc), IEEE, 2018, pp. 249–258.

[22] A. Ehsan, M. A. M. E. Abuhaliqa, C. Catal, and D. Mishra, "RESTful API testing methodologies: Rationale, challenges, and solution directions," Appl. Sci., vol. 12, no. 9, p. 4369, 2022.

[23] A. Shabbir, A. Deraman, M. Nor Bin Hassan, K. U. Sarker, and S. Kamal, "Enhancing API Quality: A Comprehensive Review of Non-Functional Requirements for Quality-centric Framework Development," KSII Trans. INTERNET Inf. Syst. ISSN 1976-7277, 2025.

[24] A. Svensson, "What is the best API from adeveloper's perspective?: Investigation of API development with fintechdevelopers in the spotlight," 2024.

[25] W. Granli, J. Burchell, I. Hammouda, and E. Knauss, "The driving forces of API evolution," in Proceedings of the 14th International Workshop on Principles of Software Evolution, 2015, pp. 28–37.

[26] J. Peddie, "Application Program Interface (API)," in The History of the GPU - Eras and Environment, Cham: Springer International Publishing, 2022.

[27] N. Kratzke, "A brief history of cloud application architectures," Appl. Sci., vol. 8, no. 8, p. 1368, 2018.

[28] T. Shimosawa, "Quality is not an act, it is a habit—Aristotle," Hypertens. Res., vol. 46, no. 5, pp. 1221–1226, May 2023.

[29] J. Ofoeda, R. Boateng, and J. Effah, "Application Programming Interface (API) Research," Int. J. Enterp. Inf. Syst., vol. 15, no. 3, pp. 76–95, Jul. 2019.

[30] M. Raatikainen, E. Kettunen, A. Salonen, M. Komssi, T. Mikkonen, and T. Lehtonen, "State of the Practice in Application Programming Interfaces (APIs): A Case Study," in European Conference on Software Architecture, Springer, 2021.

[31] B. Jin, S. Sahni, and A. Shevat, Designing Web APIs: Building APIs That Developers Love. " O'Reilly Media, Inc.," 2018.

[32] Stephen J. Bigelow, "5 stages of an API lifecycle explained," TechTarget.

[33] J. Eckhardt, A. Vogelsang, and D. M. Fernández, "Are" non-functional" requirements really non-functional? an investigation of non-functional requirements in practice," in Proceedings of the 38th international conference on software engineering, 2016, pp. 832–842.

[34] W. Behutiye, P. Karhapää, D. Costal, M. Oivo, and X. Franch, "Non-functional requirements documentation in agile software development: challenges and solution proposal," in International conference on product-focused software process improvement, Springer, 2017, pp. 515–522.

[35] J. Alvarado-Uribe, A. Y. Barrera-Animas, M. Gonzalez-Mendoza, A. L. Garcia-Gamboa, and N. Hernandez-Gress, "Towards a Standardized Evaluation of APIs Non-Functional Requirements Focused on Completeness and Soundness Qualities," Comput. y Sist., vol. 27, no. 4, pp. 889–897, 2023.

[36] N. Krishnamurthy and A. Saran, Building software: a practitioner's guide. Auerbach Publications, 2007.

[37] G. J. Singh, "10 Non-functional requirements for API design."

[38] A. C. da Silva Andrade, J. L. Braga, A. L. de Castro Leal, and F. H. Zaidan, "Risk management in software projects: an approach based on non-functional requirements," Sist. Gestão, vol. 14, no. 2, pp. 188–196, 2019.

[39] H. Ahmed, S. Zehra, U. Faseeha, H. J. Syed, and F. Samad, "Observability in Microservices: An In-Depth Exploration of Frameworks, Challenges, and Deployment Paradigms," IEEE Access, vol. 13, pp. 72011–72039, 2025.

[40] I. Gorton, L. Fong-Jones, and A. Larsson, "Observability Q&A," IEEE Softw., vol. 41, pp. 50–54, 2024.

[41] A. Chandrasehar, "The Role of Observability in Modern Software Development Lifecycle," Int. J. Sci. Res., 2021.

[42] B. Balakrishna, "Optimizing Observability: A Deep Dive into AWS Lambda Metrics," J. Artif. Intell. & Cloud Comput., 2022.

[43] N. Kratzke, "Cloud-Native Observability: The Many-Faceted Benefits of Structured and Unified Logging - A Multi-Case Study," Futur. Internet, vol. 14, p. 274, 2022.

[44] B. Sharma and D. Nadig, "eBPF-Enhanced Complete Observability Solution for Cloud-native Microservices," ICC 2024 - IEEE Int. Conf. Commun., pp. 1980–1985, 2024.

[45] R. Dhall, "Designing Graceful Degradation in Software Systems," pp. 171–179, 2017.

[46] S. Chu, J. Koe, D. Garlan, and E. Kang, "Integrating Graceful Degradation and Recovery Through Requirement-Driven Adaptation," 2024 IEEE/ACM 19th Symp. Softw. Eng. Adapt. Self-Managing Syst., pp. 122–132, 2024.

[47] D. Kumar, A. Kumar, and L. Singh, "Non-functional requirements elicitation in agile base models," Webology, vol. 19, no. 1, pp. 1992–2018, 2022.

[48] A. A. Khan, J. A. Khan, M. A. Akbar, P. Zhou, and M. Fahmideh, "Insights into software development approaches: mining Q &A repositories," Empir. Softw. Eng., vol. 29, no. 1, p. 8, 2024.

[49] J. Zou, L. Xu, M. Yang, X. Zhang, and D. Yang, "Towards comprehending the non-functional requirements through developers' eyes: An exploration of stack overflow using topic analysis," Inf. Softw. Technol., vol. 84, pp. 19–32, 2017.

[50] I. Gorton, "Software quality attributes," Essent. Softw. Archit., pp. 23–39, 2006.

[51] D. Bermbach and E. Wittern, "Benchmarking web api quality," in Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings 16, Springer, 2016, pp. 188–206.

[52] M. R. M. Assis and L. F. Bittencourt, "A survey on cloud federation architectures: Identifying functional and non-functional properties," J. Netw. Comput. Appl., vol. 72, pp. 51–71, 2016.

[53] A. Mahmoud, "An information theoretic approach for extracting and tracing non-functional requirements," in 2015 IEEE 23rd International Requirements Engineering Conference (RE), IEEE, 2015, pp. 36–45.

[54] B. De, "Api architecture trends in 2023," in API Management: An Architect's Guide to Developing and Managing APIs for Your Organization, Springer, 2023, pp. 385–411.

[55] L. Viviani, E. Guerra, J. Melegati, and X. Wang, "An empirical study about the instability and uncertainty of non-functional requirements," in International Conference on Agile Software Development, Springer Nature Switzerland Cham, 2023, pp. 77–93.

[56] S. Das, N. Deb, N. Chaki, and A. Cortesi, "Minimising conflicts among run‐time non‐functional requirements within DevOps," Syst. Eng., vol. 27, no. 1, pp. 177‐198, 2024.

[57] S. Santos, T. Pimentel, F. G. Rocha, and M. Soares, "Using Behavior-Driven Development (BDD) for Non-functional Requirements," 2024.

[58] A. Muhammad, A. Siddique, M. Mubasher, A. Aldweesh, and Q. N. Naveed, "Prioritizing non-functional requirements in agile process using multi criteria decision making analysis," IEEE Access, vol. 11, pp. 24631–24654, 2023.

[59] S. Rahy and J. M. Bass, "Managing non‐functional requirements in agile software development," IET Softw., vol. 16, no. 1, pp. 60‐72, 2022.

[60] H. M. A. El Sameaa, N. A. abd el Azim, and N. Ramadan, "Challenges of Non-functional Requirements Extraction in Agile Software Development using Machine Learning," Int. J. Comput. Appl., vol. 183, no. 43, pp. 23–26, 2021.

[61] M. Mathijssen, M. Overeem, and S. Jansen, "Identification of practices and capabilities in API management: a systematic literature review," arXiv Prepr. arXiv2006.10481, 2020.

[62] G. G. Martinez, Á. F. Del Carpio, and L. N. Gómez, "A model for detecting conflicts and dependencies in non-functional requirements using scenarios and use cases," in 2019 XLV Latin American Computing Conference (CLEI), IEEE, 2019, pp. 1–8.

[63] C. F. Castro et al., "Towards a conceptual framework for decomposing non-functional requirements of business process into quality of service attributes," in ICEIS 2019-Proceedings of the 21st International Conference on Enterprise Information Systems, SciTePress, 2019, pp. 481–492.

[64] E. Sherif, W. Helmy, and G. H. Galal-Edeen, "Proposed framework to manage non-functional requirements in agile," IEEE access, vol. 11, pp. 53995–54005, 2023.

[65] S. Pargaonkar, "A comprehensive review of performance testing methodologies and best practices: software quality engineering," Int. J. Sci. Res., vol. 12, no. 8, pp. 2008–2014, 2023.

[66] R. Koçi, X. Franch, P. Jovanovic, and A. Abelló, "Web api evolution patterns: A usage-driven approach," J. Syst. Softw., vol. 198, p. 111609, 2023.

[67] M. Lamothe, Y.-G. Guéhéneuc, and W. Shang, "A systematic review of API evolution literature," ACM Comput. Surv., vol. 54, no. 8, pp. 1–36, 2021.

[68] I. Rauf, E. Troubitsyna, and I. Porres, "A systematic mapping study of API usability evaluation methods," Comput. Sci. Rev., vol. 33, pp. 49–68, 2019.

[69] J. Scheibmeir and Y. Malaiya, "An API development model for digital twins," in 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), IEEE, 2019, pp. 518–519.

[70] Y. Liu, A. Hamou-Lhadj, J. Li, and Q. Lu, "Observability and Explainability for Software Systems Decision Making," IEEE Softw., vol. 41, no. 1, pp. 45–49, 2023.

[71] M. Saunders, P. Lewis, and A. Thornhill, Research methods for business students. Pearson education, 2009.

[72] W. G. Zikmund, B. J. Babin, J. C. Carr, and M. Griffin, "Business Research Method 8th ed," 2010, Cengage Learning.

[73] D. F. Polit and C. T. Beck, "The content validity index: are you sure you know what's being reported? Critique and recommendations," Res. Nurs. Health, vol. 29, no. 5, pp. 489–497, 2006.

[74] M. Khan, M. A. Akbar, and J. Kasurinen, "Integrating Large Language Models in Software Engineering Education: A Pilot Study through GitHub Repositories Mining," arXiv Prepr. arXiv2509.04877, 2025.

[75] M. A. Hertzog, "Considerations in determining sample size for pilot studies," Res. Nurs. Health, vol. 31, no. 2, pp. 180–191, 2008.

[76] M. Tavakol and R. Dennick, "Making sense of Cronbach's alpha," Int. J. Med. Educ., vol. 2, p. 53, 2011.

[77] F. J. Fowler Jr, Survey research methods. Sage publications, 2013.

[78] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, Experimentation in software engineering, vol. 236. Springer, 2012.

[79] M. Kasunic, "Designing an effective survey," 2005, Citeseer.

[80] M. R. Lynn, "Determination and quantification of content validity," Nurs. Res., vol. 35, no. 6, pp. 382–386, 1986.

[81] A. A. S. Almohanna, K. T. Win, S. Meedya, and E. Vlahu-Gjorgievska, "Design and content validation of an instrument measuring user perception of the persuasive design principles in a breastfeeding mHealth app: A modified Delphi study," Int. J. Med. Inform., vol. 164, p. 104789, 2022.

[82] K. C. Pentapati, D. Chenna, V. S. Kumar, and N. Kumar, "Reliability generalization meta-analysis of Cronbach's alpha of the oral impacts on daily performance (OIDP) questionnaire," BMC Oral Health, vol. 25, no. 1, p. 220, 2025.

[83] H. N. Boone Jr and D. A. Boone, "Analyzing likert data," J. Ext., vol. 50, no. 2, p. 48, 2012.

[84] M. Perepletchikov, C. Ryan, K. Frampton, and Z. Tari, "Coupling metrics for predicting maintainability in service-oriented designs," in 2007 Australian Software Engineering Conference (ASWEC'07), IEEE, 2007, pp. 329–340.

[85] E. dos Santos and S. Casas, "An API Management Software Quality Metamodel based on Square and GQM," Eng. e soluç oes Ciência e Tecnol. para o Desenvolv. Hum., pp. 249–266, 2025.

[86] N. Dragoni et al., "Microservices: yesterday, today, and tomorrow," Present ulterior Softw. Eng., pp. 195–216, 2017.

Perceived Usefulness and Perceived Ease of Use as Predictors of Attitude Toward IoT Adoption Among Rice Farmers

Hermin Arrang[1]*, Sek Yong Wee[2]*, Nazrulazar Bin Bahaman[3], Jack Febrian Rusdi[4]
Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia [1, 2, 3]
Department of Informatics Engineering, Universitas Kristen Indonesia Paulus, Makassar, Indonesia [1]
Department of Informatics Engineering, Universitas Teknologi Bandung, Bandung, Indonesia [4]

***Abstract*—This study investigates key drivers influencing rice farmers' attitudes toward Internet of Things (IoT) adoption in Indonesia, using the Technology Acceptance Model (TAM) as an analytical lens. Specifically, it evaluates the predictive roles of Perceived Usefulness (PU) and Perceived Ease of Use (PEOU), both of which are posited to shape users' Attitude Toward Usage (ATT). Survey data were obtained from 62 smallholder farmers in Bandung Regency and examined through Partial Least Squares Structural Equation Modeling (PLS-SEM). The study confirms that both PU and PEOU significantly contribute to the formation of favorable farmer attitudes, with the model showing high explanatory strength (R^2 = 0.723). PU captures the perceived benefit of IoT for improving productivity and efficiency, while PEOU reflects user-friendly design and ease of integration into existing agricultural routines. These findings extend TAM's validity in rural, low-tech farming contexts and offer actionable insights for technology developers, government agencies, and agricultural organizations seeking to foster digital transformation. By confirming the relevance of usability and perceived value, this study supports targeted design and communication strategies that align with farmers' expectations. It also lays the groundwork for broader ASEAN agricultural resilience efforts by emphasizing inclusive technology pathways. Future research may incorporate sociocultural dimensions and systemic barriers to expand the model's applicability in diverse farming environments.**

Keywords*—IoT; attitude towards IoT; Perceived Usefulness; Perceived Ease of Use; technology adoption

I. Introduction

The Internet of Things (IoT) holds considerable promise in transforming the agricultural sector by improving efficiency, productivity, and sustainability [1]. In the context of rice farming, IoT applications include monitoring environmental conditions, optimizing irrigation, and delivering real-time data to support informed decision-making [2]. Despite these advantages, IoT adoption among farmers remains limited, and gaining insight into the factors influencing their attitudes is essential for effective implementation [3], [4].

Attitude plays a central role in shaping an individual's intention and behavior toward adopting new technology. The Technology Acceptance Model (TAM), a commonly utilized framework in technology adoption research, identifies Perceived Usefulness (PU) and Perceived Ease of Use (PEOU) as key antecedents of attitude [5], [6]. PU denotes the perception that the utilization of a specific technology enhances task efficiency, whereas PEOU reflects the notion that the technology is easy to operate.

For rice farmers, cultivating a positive attitude toward IoT adaption (ATT) is a critical step in encouraging experimentation and adoption [4], [7]. Accordingly, this research explores how PU and PEOU shape farmers' attitudes within the TAM framework, focusing on rice farmers in Bandung Regency, West Java, Indonesia [8].

TAM has been widely applied across diverse technological domains, consistently validating the predictive strength of PU and PEOU in shaping attitudes toward technology [9], [10], [11]. In agriculture specifically, several studies affirm the significance of these constructs in driving the adoption of technological innovations [12], [13].

Nonetheless, prior research also acknowledges that other variables—such as individual innovativeness, trust, and social norms—may influence farmers' attitudes [12], [14]. While recognizing the multifaceted nature of technology adoption, this study purposefully concentrates on the foundational constructs within TAM.

Despite the existing body of literature, a research gap persists in understanding how PU and PEOU specifically affect rice farmers' ATT. This study addresses this gap by drawing on empirical data from rice farmers in Bandung Regency.

Accordingly, the research centers on evaluating the impact of these two constructs in shaping farmer attitudes. The central questions are formalized into the following hypotheses:

- H9: How does Perceived Usefulness (PU) influence rice farmers' attitudes toward IoT adaption (ATT)?
- H11: How does Perceived Ease of Use (PEOU) influence rice farmers' attitudes toward IoT adaption (ATT)?

This study is principally designed to investigate how PU and PEOU shape farmers' ATT. The study contributes theoretically by extending the application of TAM to agricultural IoT contexts, and practically by offering insights for stakeholders—government agencies, technology developers, and farming organizations—on how to cultivate positive farmer perceptions through targeted interventions that highlight both usefulness and usability.

*Corresponding author.

II. Literature Review

A. Internet of Things (IoT) in Agriculture

IoT is increasingly recognized as a pivotal innovation capable of reshaping multiple industries [15], [16], [17], [18], [19], [20], including agriculture [21], [22], [23]. IoT in agriculture entails deploying sensors, actuators, and other interconnected devices to gather and transmit information related to various aspects of farming operations, including environmental parameters, crop vitality, and soil characteristics. The resulting data can be processed to generate actionable knowledge for farmers, supporting more precise decision-making and enhancing agricultural efficiency.

The potential benefits of IoT in rice farming are substantial. For example, IoT sensors can monitor soil moisture levels and automate irrigation, reducing water waste and increasing crop yields [19], [21], [24], [25], [26], [27], [28], [29]. Drones integrated with IoT imaging systems can assess plant health and facilitate early detection of agricultural threats, allowing for timely intervention [30], [31], [32], [33], [34]. IoT can also assist in optimizing fertilizer use, minimizing environmental impact, and enhancing overall efficiency.

B. Technology Acceptance Model (TAM)

TAM is among the predominant theoretical frameworks for understanding how individuals accept and adopt information technologies [35], [36], [37]. According to TAM, a person's attitude toward utilizing a technology is shaped by two central perceptions: Perceived Usefulness (PU), which represents the notion that the technology contributes to improved job performance [38], [39], and PEOU, which embodies the notion that the system requires minimal effort to operate [10].

TAM argues that PU and PEOU positively influence ATT [40], which subsequently shapes the behavioral tendency toward adopting the technology [41]. Within the scope of this research, TAM serves as a relevant framework for examining how rice farmers engage with and embrace IoT technology. The questionnaire employed in the present work assesses PU and PEOU using measurement scales derived from previously validated TAM instruments. Although TAM has been expanded with various external constructs in prior studies, the current analysis concentrates on the fundamental linkage between PU and PEOU as principal predictors of ATT.

C. Perceived Usefulness and Attitude Towards IoT

PU is a significant predictor of ATT [10]. if they perceive improvements in task efficiency or outcome quality. [42]. In the context of rice farming, farmers may perceive IoT as useful if it can help them to increase crop yields, reduce input costs (e.g., water, fertilizer), save time and labor, and make better-informed decisions [43].

Prior research has demonstrated that PU significantly influences ATT of various agricultural technologies, such as farm information systems and precision agriculture technologies [44]. Therefore, this study hypothesizes (H9) that PU positively influences rice farmers' ATT.

D. Perceived Ease of Use and Attitude Toward IoT

PEOU is also an important factor in shaping ATT [10]. Individuals tend to have a positive ATT if they perceive it as easy to learn, easy to use, and requiring minimal mental effort [45].

In the context of rice farming, farmers may perceive IoT as easy to use if the user interface is intuitive, the system is reliable and stable, training and support are readily available, and the technology is compatible with existing farming practices [46], [47].

Previous research has found that PEOU significantly affects ATT across different domains, including agriculture [14], [48]. Hence, this study hypothesizes (H11) that PEOU positively influences rice farmers' ATT.

E. Attitude Towards IoT and Adoption Intention

ATT represents a person's general assessment of the IoT technology, reflecting their positive or negative feelings about it. A positive attitude toward technology is likely to lead to a stronger intention to adopt and use it [14], [49].

In this study, rice farmers' ATT refers to their feelings and evaluations regarding the use of IoT in their farming practices. The questionnaire used in this research measures farmers' ATT using items that assess their feelings about using IoT for collaborative farming [50], [51].

Prior research consistently demonstrates a robust association between attitude toward technology and adoption intention. Therefore, this study hypothesizes that rice farmers' positive attitude toward IoT enhances their behavioral intention toward adopting the technology.

F. Conceptual Framework

Framework Representing TAM and Associated Determinants as presented in Fig. 1.

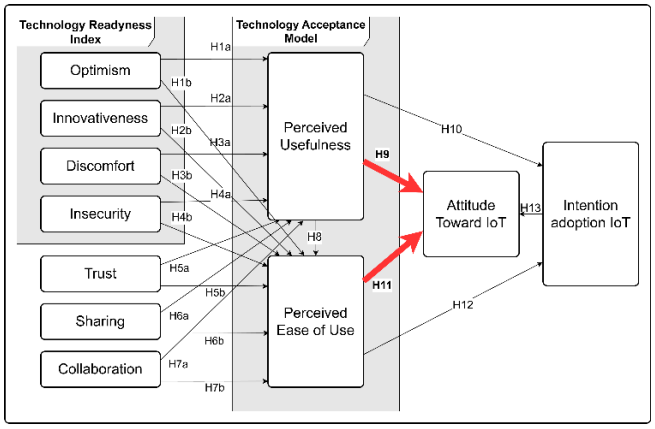


Fig. 1. Framework representing the technology acceptance model and influencing factors.

Fig. 1 illustrates the analytical model developed for this research. This model is based on the TAM, which posits that an individual's intention to embrace a technology is influenced by their ATT. That attitude, in turn, is principally guided by two core constructs: PU and PEOU. In this study, we specifically examine the relationships between PU, PEOU, ATT, and

Intention Adoption IoT. The framework also illustrates those other factors, such as Optimism, Innovativeness, Discomfort, Insecurity, Trust, Sharing, and Collaboration, can influence farmers' perceptions of PU and PEOU. While these factors are acknowledged, the analysis centers on the direct effects among PU (H9) and PEOU (H11) toward ATT, which constitute the fundamental constructs of the TAM framework.

III. Methodology

A. Research Design

This study employs a quantitative approach utilizing survey methodology -based method to examine the elements affecting rice farmers' ATT. A standardized questionnaire was applied to methodically gather information from farmers concerning their views and dispositions. The research process is systematically illustrated in Fig. 2, detailing the stages from conceptualization to data collection and analysis.

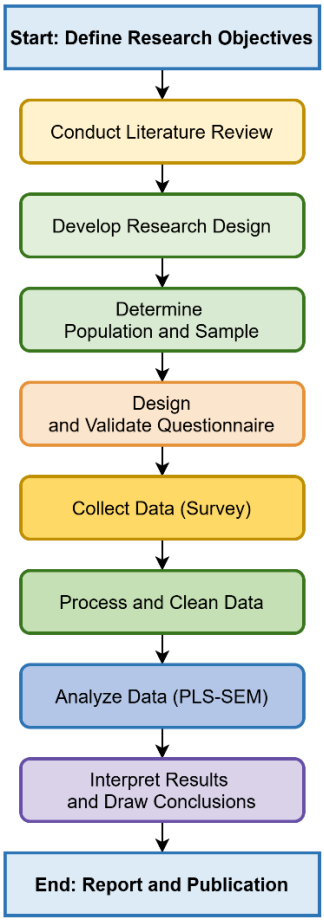


Fig. 2. Research process flowchart for IoT adoption study.

The survey method was chosen as it allows for statistical analysis of predefined variables—PU, PEOU, and ATT—and enables generalization of findings to a broader population. Given that IoT adoption in agriculture is still emerging, this approach provides empirical insights into farmers' acceptance of new technology.

B. Population and Sample

1) Target population: The population under investigation comprises rice farmers from Bandung Regency, West Java, Indonesia.

2) Sampling method: This study employed both purposive and stratified sampling techniques to ensure targeted and representative respondent selection to ensure a representative and relevant dataset:

a) Purposive Sampling was employed to select farmers who had prior exposure to IoT technology in agricultural practices. This ensured that respondents had a sufficient understanding of IoT to provide meaningful insights into its usefulness and ease of use.

b) Stratified Sampling was applied based on farmers' land size, farming experience, and education level, ensuring balanced representation across different categories of respondents.

3) Sample size determination: A total of 62 rice farmers were selected as respondents. The sample size was determined based on statistical power analysis, ensuring that it provides sufficient data for meaningful inference while maintaining computational efficiency for PLS-SEM analysis. While this sample size is appropriate for PLS-SEM, it should be noted that generalizing the findings to a broader population of rice farmers may require caution, given the inherent contextual variations.

C. Research Instrument

Primary empirical input was obtained through a standardized instrument, divided into two discrete components:

- Section 1: Demographic Information – Collected respondents' age, gender, education level, and farming experience to provide contextual insights.
- Section 2: Measurement of Study Variables – Included statements assessing PU, PEOU, and ATT, assessed through A 7-point Likert-type scale were adopted, with endpoints defined as 1 (strongly disagree) and 7 (strongly agree).

All survey indicators were derived from established scales validated in prior research on technology adoption. Respondents completed the questionnaire under the guidance of enumerators, who assisted in clarifying questions to ensure accurate responses.

D. Data Collection Procedure

Data collection was conducted through direct questionnaire distribution, with enumerators supervising the process to facilitate respondent understanding. The data collection period was February 9th to February 12th, 2025.

Before survey administration, participants received details concerning the aims of the research, assurance of response privacy, and participants' entitlement to withdraw at any phase.

Informed consent was obtained from all respondents before proceeding with data collection. The data collection procedure and analytical approach are illustrated in Fig. 1, providing a step-by-step breakdown from administration to statistical processing.

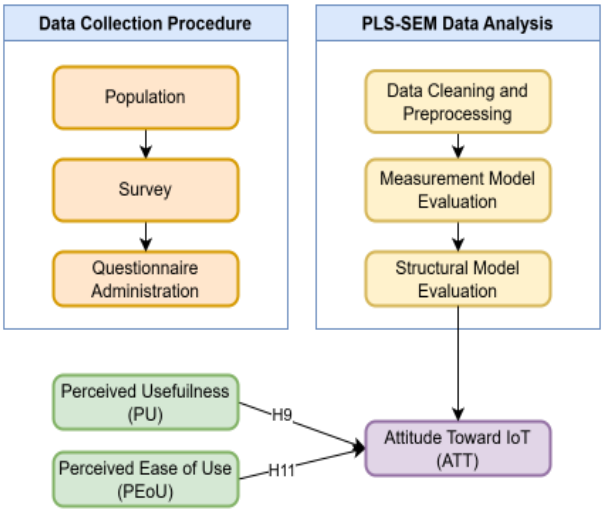


Fig. 1. Data collection and analysis framework.

This diagram illustrates the data collection procedure and analysis workflow for this study. It outlines the structured process from sampling selection to hypothesis testing, utilizing Partial Least Squares Structural Equation Modeling (PLS-SEM) to map the causal pathways linking principal constructs.

The hypotheses tested in the analysis have been updated to match the conceptual framework outlined in the Literature Review and the structure of the Results Section. Specifically, H9 and H11.

Each stage of the analysis follows a systematic approach, starting with data cleaning and preprocessing, followed by measurement model evaluation, and concluding with structural model validation and hypothesis testing using Smart PLS 3 software.

E. Data Analysis Methods

The collected data underwent analysis using both descriptive and inferential statistical techniques, as illustrated in Fig. 1. Demographic profiles of the respondents were outlined using descriptive statistical techniques respondents, including measures such as frequencies, percentages, means, and standard deviations. Furthermore, PLS-SEM was selected as the analytical technique to test the hypothesized paths between constructs, owing to its capacity to accommodate complex models and its applicability in contexts with modest sample sizes. Furthermore, its relaxed assumptions regarding data distribution made it a suitable choice for this study. All computations were carried out using SmartPLS 3. The detailed evaluation of the measurement and structural models, including criteria and results, will be presented in the "PLS-SEM Analysis" section under Results and Discussion.

IV. Results and Discussion

The subsequent part delineates empirical outputs and interprets the observed influences on rice farmers' ATT. The results include the characteristics of the respondents, the assessment of instrument validity and reliability, along with the PLS-SEM results employed to examine the hypotheses concerning PU, PEOU, and ATT.

A. Respondent Characteristics

The data on respondent characteristics are presented to provide an overview of the participants' backgrounds and the research context. This information is relevant for interpreting the analysis results related to ATT.

Table I presents the frequency distributions of respondents based on their demographic characteristics and technology usage. The following tables present the characteristics of the 62 respondents who participated in the survey conducted between February 9th and February 12th, 2025.

TABLE I. Demographic and Farming Characteristics of Respondents

Respondent Characteristic	Categories	Frequency	Percentage
Age	29 years or younger	1	1.61%
	30-39 years	9	14.52%
	40-49 years	11	17.74%
	50-59 years	25	40.32%
	60 years or older	16	25.81%
Gender	Male	46	74.19%
	Female	16	25.81%
Rice Field Area	<0.5 hectares	19	30.65%
	0.5-1 hectare	26	41.94%
	1-2 hectares	17	27.42%
Farming Experience	1-5 years	6	9.68%
	6-10 years	6	9.68%
	11-15 years	10	16.13%
	16-20 years	17	27.42%
	21-25 years	17	27.42%
	>25 years	6	9.68%
Education Level	No formal education	1	1.61%
	Elementary school	43	69.35%
	Junior high school	15	24.19%
	Senior high school	3	4.84%
Technology Use in Farming	Never	59	95.16%
	Rarely/Occasionally	3	4.84%

B. Validity and Reliability Tests

Instrument validity was assessed to confirm that the measurement items appropriately capture the targeted constructs. In this study, the validity was tested through Pearson correlation analysis, which quantifies the strength and direction of the linear relationship between each statement item and the overall construct. A correlation coefficient above 0.3 indicates

that the item is valid and meaningfully contributes to measuring the construct.

Table II presents the revised questionnaire items used to measure each construct—PU, PEOU, ATT, and Intention to Adopt IoT. Each item was adapted from validated TAM instruments and reworded to preserve academic neutrality and consistency.

TABLE II. VALIDITY TEST RESULTS OF IOT ADOPTION CONSTRUCTS

Variable	Statement Item	Calculated r	Critical r	Validity
Perceived Usefulness (PU)	IoT technology is perceived to improve rice harvest outcomes.	0.987	0.3	Valid
	IoT enables discovery of efficient and modern agricultural practices.	0.989	0.3	Valid
	IoT enhances farm management capabilities.	0.99	0.3	Valid
	IoT contributes to faster task completion in rice fields.	0.974	0.3	Valid
Perceived Ease of Use (PEOU)	IoT is considered easy to operate in rice farming.	0.828	0.3	Valid
	IoT is perceived as easily integrable into farming routines.	0.912	0.3	Valid
	Functions of IoT in farming are easy to comprehend.	0.907	0.3	Valid
	IoT technology adapts well to individual farming needs.	0.919	0.3	Valid
Attitude Toward IoT adaption (ATT)	IoT facilitates collaborative farming to improve yields.	0.947	0.3	Valid
	Farmers are inclined to share experiences via IoT platforms.	0.926	0.3	Valid
	IoT is preferred over traditional coordination methods.	0.96	0.3	Valid
	Enthusiasm exists for using IoT to support agricultural cooperation.	0.955	0.3	Valid
Intention to Adopt IoT	Farmers plan to apply IoT in future rice farming.	0.984	0.3	Valid
	IoT is seen as instrumental for increasing productivity.	0.984	0.3	Valid
	Interest exists in exploring diverse IoT applications for farming.	0.986	0.3	Valid
	Farmers intend to recommend IoT if proven beneficial.	0.979	0.3	Valid

Table II present the validity test results for four key variables: PU, PEOU, ATT, and Intention to Adopt IoT. Each statement item within these constructs was examined to verify its reliability in capturing respondent perceptions. The correlation coefficients for all items exceed the critical threshold of 0.3, confirming their validity as research indicators.

The validation results underscore the robustness of the instrument used in this study, ensuring that the measured variables accurately reflect respondents' perceptions and attitudes. This validation strengthens the subsequent analysis, enabling precise interpretation of IoT adoption tendencies within the agricultural sector.

Reliability assessment employed Cronbach's Alpha as a metric for internal consistency, adopting 0.6 as the minimum acceptable threshold.

Table III displays the outcomes of the reliability assessment, showing that all constructs exhibit acceptable internal consistency ($\alpha > 0.6$).

TABLE III. RELIABILITY TEST RESULTS

Variable	Reliability Index	Critical Value	Result
Perceived Usefulness	0.989	0.6	Reliable
Perceived Ease of Use	0.912	0.6	Reliable
Attitude Toward IoT	0.959	0.6	Reliable
Intention Adoption IoT	0.988	0.6	Reliable

C. PLS-SEM Analysis

Multiple statistical procedures were applied to confirm the reliability and validity of the measurement tool.

1) Convergent validity: Convergent validity was assessed to determine whether indicators effectively measure their respective constructs. Factor loadings and Average Variance Extracted (AVE) were used to evaluate this, with thresholds suggesting validity are factor loadings > 0.6 and AVE > 0.5, indicating strong construct validity (Table IV).

TABLE IV. LOADING FACTOR AND AVE RESULTS

Construct	Indicator	Factor Loading	Critical Value	AVE
Attitude Toward IoT	ATT1	0.951	0.6	0.897
	ATT2	0.937	0.6	
	ATT3	0.954	0.6	
	ATT4	0.947	0.6	
Perceived Ease of Use	PEOU1	0.852	0.6	0.793
	PEOU2	0.89	0.6	
	PEOU3	0.885	0.6	
	PEOU4	0.934	0.6	
Perceived Usefulness	PU1	0.987	0.6	0.97
	PU2	0.99	0.6	
	PU3	0.99	0.6	
	PU4	0.974	0.6	

2) Discriminant validity: It serves to verify that no latent variable overlaps significantly with another, maintaining construct independence. The cross-loading test was performed to verify whether indicators correlate more strongly with their designated construct than with other constructs (Table V).

TABLE V. CROSS-LOADING RESULTS

Attitude Toward IoT	Perceived Ease of Use	Perceived Usefulness	Result
ATT1	0.951	0.777	0.786
ATT2	0.937	0.801	0.795
ATT3	0.954	0.719	0.782
ATT4	0.947	0.705	0.752
PEOU1	0.785	0.852	0.698
PEOU2	0.535	0.89	0.5
PEOU3	0.50. 5	0.885	0.472
PEOU4	0.687	0.934	0.651
PU1	0.761	0.698	0.987
PU2	0.76	0.5	0.99
PU3	0.776	0.472	0.99
PU4	0.826	0.651	0.974

3) Reliability test: The reliability coefficients derived from Cronbach's Alpha and Composite Reliability, where values above 0.7 are considered indicative of robust internal consistency (Table VI).

TABLE VI. RELIABILITY TEST RESULTS

Latent Variable	Average Variance Extracted (AVE)	Critical Value	Criterion (AVE > 0.5)
Attitude Toward IoT	0.897	0.5	Valid
Perceived Ease of Use	0.793	0.5	Valid
Perceived Usefulness	0.97	0.5	Valid

4) Structural model evaluation: Structural model evaluation was conducted to examine inter-construct linkages through multiple analytical indicators.

R-Square (R^2) quantifies how predictor variables explain fluctuations within the target construct (Table VII).

TABLE VII. R-SQUARE RESULTS

Dependent Variable	R-Square	Strength of relationship
Attitude Toward IoT	0.723	Strong

Effect Size (f-Square) measures the impact of the influence of predictor variables on the outcome variable (Table VIII).

TABLE VIII. F-SQUARE EFFECT SIZE

Independent Variable	Attitude Toward IoT
Perceived Ease of Use	0.337
Perceived Usefulness	0.732

Predictive Relevance (Q^2). Predictive relevance (Q^2) was tested to measure the predictive capacity of the model future outcomes (Table IX).

TABLE IX. Q^2 PREDICTIVE RELEVANCE RESULTS

Dependent Variable	SSO	SSE	Q^2 Value
Attitude Toward IoT	248	91.378	0.632

5) Hypothesis testing: Hypothesis testing was conducted to determine whether the proposed relationships between constructs reach statistical significance, indicating that bot PU and PEOU significantly influence ATT (Table X).

TABLE X. PATH COEFFICIENTS AND P-VALUES

Hypothesis	Variable	Original sample (O)	T-statistics (|O/STDEV|)	P-values (5%)	Result
H9	Perceived Usefulness -> Attitude Toward IoT	0.564	4,613	<0.001	Supported
H11	Perceived Ease of Use -> Attitude Toward IoT	0.382	3,454	0.001	Supported

The results in Table X confirm that both hypotheses are statistically supported. For H9, the relationship between PU and ATT, the path coefficient is 0.564, accompanied by a T-statistic of 4.613 and a P-value below 0.001, indicating a statistically significant relationship. Since the T-statistic (4.613) surpasses the established benchmark of 1.96 and the P-value (<0.001) falls below the 0.05 significance level, the proposed hypothesis is confirmed by empirical evidence. This finding confirms that PU exerts a significant positive effect on rice farmers' ATT.

Similarly, H11 is supported by a path coefficient of 0.382, a T-statistic of 3.454, and a P-value of 0.001—each meeting the criteria for statistical significance ($T > 1.96$; $P < 0.05$). The data confirm that PEOU has a meaningful and beneficial influence on rice farmers' ATT. This confirms that PPEOU significantly and positively influences farmers' ATT. The results confirm that the relationships involving PU and PEOU are statistically supported antecedents of rice farmers' ATT, reinforcing the theoretical premises of TAM.

D. Discussion

This study provides empirical evidence on the key factors influencing rice farmers' Attitude Toward Technology (ATT) within the agricultural domain. Consistent with the TAM and prior research, PU was found to significantly influence ATT. When farmers perceive IoT as a beneficial tool for enhancing productivity—such as improving crop yields, reducing operational costs, and increasing time efficiency—they are more inclined to adopt a positive stance.

Similarly, PEOU emerged as a key factor in shaping attitudes, reaffirming TAM's emphasis on usability. Farmers who regard IoT as intuitive and straightforward to use tend to demonstrate a higher propensity for adoption. This underscores the practical relevance of ergonomic system architecture,

focused capacity-building initiatives, and readily available technical assistance.

While this study primarily concentrates on PU and PEOU, it is essential to recognize that additional variables—such as cultural context, social influence, and personal innovativeness—may also affect farmers' technology perceptions. Future research could extend the model to include these complementary dimensions.

Within the scope of this analysis, however, our focus remains on the central constructs grounded in TAM. The hypotheses tested (H9 and H11) confirm the significant impact of PU and PEOU as principal predictors of ATT, supporting the robustness of the conceptual framework.

V. Conclusion

This study analyzed the factors influencing rice farmers' attitudes toward IoT adoption, emphasizing Perceived Usefulness (PU) and Perceived Ease of Use (PEOU).

A. Key Findings

- PU significantly influences farmers' attitudes toward IoT adoption. Farmers who recognize IoT as beneficial for improving farming practices—such as enhancing crop yields and efficiency — have an increased propensity to form a positive attitude toward the technology.
- PEOU significantly impacts farmers' attitudes toward IoT adoption. The usability of IoT systems plays a crucial role in adoption. Farmers who find IoT easy to use are more inclined to exhibit a favorable perspective, reinforcing the importance of designing intuitive technologies.
- While the conceptual model includes other variables such as Optimism, Innovativeness, Discomfort, and others, this study specifically did not test their direct influence on Attitude Toward IoT. Therefore, these findings do not provide a conclusion on the influence of these variables, but rather confirm that PU and PEOU serve as the principal predictors of IoT adoption among rice farmers based on the scope and focus of this research.

B. Implications for IoT Adoption in Agriculture

- Emphasizing Benefits and Ease of Use, IoT adoption initiatives should clearly communicate practical benefits, such as increased productivity, cost savings, and time efficiency, while ensuring technology remains accessible and easy to use.
- User-Centered IoT Development IoT systems should incorporate intuitive interfaces and provide adequate training, enabling farmers to seamlessly integrate technology into their agricultural practices.
- Future Research Directions: Further studies should explore contextual factors, such as social norms, cultural influences, policy frameworks, and peer networks, which may shape farmers' attitudes toward IoT adoption.

C. Final Conclusion

This study concludes that rice farmers' attitudes toward IoT adoption are primarily shaped by PU and PEOU, highlighting the need for user-centric technological interventions in agriculture. Therefore, successful IoT implementation in agriculture must prioritize solutions that deliver clear benefits while remaining user-friendly. By ensuring accessibility and demonstrating tangible advantages, technology developers and policymakers can create an environment that fosters IoT adoption and enhances farming productivity.

Acknowledgment

The authors sincerely appreciate all respondents for their participation in this study. Their insights provided crucial data for the analysis. Additionally, gratitude is extended to those who assisted in data processing, whose support ensured accuracy in statistical computations and interpretation.

References

[1] A. L. Duguma and X. Bai, "How the internet of things technology improves agricultural efficiency," Artif Intell Rev, vol. 58, no. 2, pp. 1–26, Feb. 2025, doi: 10.1007/S10462-024-11046-0/TABLES/5.

[2] P. Bamurigire, A. Vodacek, A. Valko, and S. R. Ngoga, "Simulation of Internet of Things Water Management for Efficient Rice Irrigation in Rwanda," Agriculture 2020, Vol. 10, Page 431, vol. 10, no. 10, p. 431, Sep. 2020, doi: 10.3390/AGRICULTURE10100431.

[3] R. Fragomeli, A. Annunziata, and G. Punzo, "Promoting the Transition towards Agriculture 4.0: A Systematic Literature Review on Drivers and Barriers," Sustainability (Switzerland) , vol. 16, no. 6, p. 2425, Mar. 2024, doi: 10.3390/SU16062425/S1.

[4] Y. Shi, A. B. Siddik, M. Masukujjaman, G. Zheng, M. Hamayun, and A. M. Ibrahim, "The Antecedents of Willingness to Adopt and Pay for the IoT in the Agricultural Industry: An Application of the UTAUT 2 Theory," Sustainability 2022, Vol. 14, Page 6640, vol. 14, no. 11, p. 6640, May 2022, doi: 10.3390/SU14116640.

[5] L. G. Wallace and S. D. Sheetz, "The adoption of software measures: A technology acceptance model (TAM) perspective," Information & Management, vol. 51, no. 2, pp. 249–259, Mar. 2014, doi: 10.1016/J.IM.2013.12.003.

[6] R. Scherer, F. Siddiq, and J. Tondeur, "The technology acceptance model (TAM): A meta-analytic structural equation modeling approach to explaining teachers' adoption of digital technology in education," Comput Educ, vol. 128, pp. 13–35, Jan. 2019, doi: 10.1016/J.COMPEDU.2018.09.009.

[7] N. Mishra et al., "Technology in farming: Unleashing farmers' behavioral intention for the adoption of agriculture 5.0," PLoS One, vol. 19, no. 8, p. e0308883, Aug. 2024, doi: 10.1371/JOURNAL.PONE.0308883.

[8] A. H. Tanjung et al., "Flypaper effect assessment methods in the expansion of regional autonomy," MethodsX, vol. 8, p. 101387, Jan. 2021, doi: 10.1016/j.mex.2021.101387.

[9] R. J. Thomas, G. O'Hare, and D. Coyle, "Understanding technology acceptance in smart agriculture: A systematic review of empirical research in crop production," Technol Forecast Soc Change, vol. 189, p. 122374, Apr. 2023, doi: 10.1016/J.TECHFORE.2023.122374.

[10] F. Abdullah, R. Ward, and E. Ahmed, "Investigating the influence of the most commonly used external variables of TAM on students' Perceived Ease of Use (PEOU) and Perceived Usefulness (PU) of e-portfolios," Comput Human Behav, vol. 63, pp. 75–90, Oct. 2016, doi: 10.1016/J.CHB.2016.05.014.

[11] N. Shafie, S. Salam, S. F. A. Fesol, and J. F. Rusdi, "Factors influencing technology acceptance for ubiquitous public transportation services in tourism," Bulletin of Electrical Engineering and Informatics, vol. 12, no. 6, pp. 3869–3879, Dec. 2023, doi: 10.11591/eei.v12i6.5822.

[12] B. Mckenney and W. Gottwald, "The Impact of Perceived Usefulness and Perceived Ease of Use on IoT Acceptance Among Information

Technology Professionals: A Correlational Study," Capella University ProQuest Dissertations & Theses, 2021.

[13] R. Ninsiima, P. Mshenga, and D. Okello, "Determinants of smallholder barley farmers' intentions to adopt blockchain technology: a Technology Acceptance Model approach in Uganda," Front Sustain Food Syst, vol. 9, p. 1552637, Mar. 2025, doi: 10.3389/FSUFS.2025.1552637/BIBTEX.

[14] H. Dong, H. Wang, and J. Han, "Understanding Ecological Agricultural Technology Adoption in China Using an Integrated Technology Acceptance Model—Theory of Planned Behavior Model," Front Environ Sci, vol. 10, p. 927668, Jun. 2022, doi: 10.3389/FENVS.2022.927668/BIBTEX.

[15] J. F. Rusdi et al., "Reporting of Hospital Facility on Smartphone," in Journal of Physics: Conference Series, IOP Publishing Ltd, Apr. 2021, p. 012013. doi: 10.1088/1742-6596/1807/1/012013.

[16] J. F. Rusdi et al., "A Tourist Tracking Model by Tourist Bureau," in 2019 International Conference Of Science and Information Technology in Smart Administration (ICSINTeSA), 2019.

[17] R. Hardi, N. Suryana, N. C. Pee, A. S. Pribadi, J. F. Rusdi, and A. Junaidi, "The Object Tracking System at the Service Delivery Center of the Traveling Salesperson Problem Method," in Journal of Physics: Conference Series, IOP Publishing Ltd, Apr. 2021, p. 12034. doi: 10.1088/1742-6596/1807/1/012034.

[18] J. F. Rusdi, A. Nurhayati, H. Gusdevi, M. I. Fathulloh, A. Priyono, and R. Hardi, "IoT-based Covid-19 Patient Service Robot Design," 3rd International Conference on Cybernetics and Intelligent Systems, ICORIS 2021, 2021, doi: 10.1109/ICORIS52787.2021.9649461.

[19] J. F. Rusdi et al., "Field Reporting Irrigation System via Smartphone," in Journal of Physics: Conference Series, IOP Publishing Ltd, Apr. 2021, p. 12012. doi: 10.1088/1742-6596/1807/1/012012.

[20] J. F. Rusdi, M. Naseer, B. Pudjoatmodjo, S. A. Sidik, I. N. Aziz, and R. Hardi, "Motor Cycle SmartKey System," in 2021 3rd International Conference on Cybernetics and Intelligent System (ICORIS), Institute of Electrical and Electronics Engineers (IEEE), Dec. 2021, pp. 1–5. doi: 10.1109/ICORIS52787.2021.9649494.

[21] M. S. Farooq, S. Riaz, A. Abid, T. Umer, and Y. Bin Zikria, "Role of IoT Technology in Agriculture: A Systematic Literature Review," Electronics 2020, Vol. 9, Page 319, vol. 9, no. 2, p. 319, Feb. 2020, doi: 10.3390/ELECTRONICS9020319.

[22] J. Xu, B. Gu, and G. Tian, "Review of agricultural IoT technology," Artificial Intelligence in Agriculture, vol. 6, pp. 10–22, Jan. 2022, doi: 10.1016/J.AIIA.2022.01.001.

[23] W. S. Kim, W. S. Lee, and Y. J. Kim, "A Review of the Applications of the Internet of Things (IoT) for Agricultural Automation," Journal of Biosystems Engineering, vol. 45, no. 4, pp. 385–400, Dec. 2020, doi: 10.1007/S42853-020-00078-3/METRICS.

[24] J. N. Ndunagu, K. E. Ukhurebor, M. Akaaza, and R. B. Onyancha, "Development of a Wireless Sensor Network and IoT-based Smart Irrigation System," Appl Environ Soil Sci, vol. 2022, no. 1, p. 7678570, Jan. 2022, doi: 10.1155/2022/7678570.

[25] E. Bwambale, F. K. Abagale, and G. K. Anornu, "Smart irrigation monitoring and control strategies for improving water use efficiency in precision agriculture: A review," Agric Water Manag, vol. 260, p. 107324, Feb. 2022, doi: 10.1016/J.AGWAT.2021.107324.

[26] S. Touil, A. Richa, M. Fizir, J. E. Argente García, and A. F. Skarmeta Gómez, "A review on smart irrigation management strategies and their effect on water savings and crop yield," Irrigation and Drainage, vol. 71, no. 5, pp. 1396–1416, Dec. 2022, doi: 10.1002/IRD.2735.

[27] R. Nageswara Rao and B. Sridhar, "IoT based smart crop-field monitoring and automation irrigation system," Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018, pp. 478–483, Jun. 2018, doi: 10.1109/ICISC.2018.8399118.

[28] K. Obaideen et al., "An overview of smart irrigation systems using IoT," Energy Nexus, vol. 7, p. 100124, Sep. 2022, doi: 10.1016/J.NEXUS.2022.100124.

[29] F. Mulyawati, I. Sudarsono, and J. F. Rusdi, "Braided River Irrigation used Siphon Method," IOP Conf Ser Earth Environ Sci, vol. 955, no. 1, p. 012001, Jan. 2022, doi: 10.1088/1755-1315/955/1/012001.

[30] J. F. Rusdi, S. Salam, N. A. Abu, S. Sahib, M. Naseer, and A. A. Abdullah, "Drone Tracking Modelling Ontology for Tourist Behavior," J Phys Conf Ser, vol. 1201, no. 1, p. 012032, May 2019, doi: 10.1088/1742-6596/1201/1/012032.

[31] C. Singh, R. Mishra, H. P. Gupta, and P. Kumari, "The Internet of Drones in Precision Agriculture: Challenges, Solutions, and Research Opportunities," IEEE Internet of Things Magazine, vol. 5, no. 1, pp. 180–184, Mar. 2022, doi: 10.1109/IOTM.006.2100100.

[32] P. K. Dutta and S. Mitra, "Application of Agricultural Drones and IoT to Understand Food Supply Chain During Post COVID-19," Agricultural Informatics: Automation Using the IoT and Machine Learning, pp. 67–87, Jan. 2021, doi: 10.1002/9781119769231.CH4.

[33] A. Nayyar, B. Le Nguyen, and N. G. Nguyen, "The Internet of Drone Things (IoDT): Future Envision of Smart Drones," Advances in Intelligent Systems and Computing, vol. 1045, pp. 563–580, 2020, doi: 10.1007/978-981-15-0029-9_45.

[34] A. K. Saha et al., "IOT-based drone for improvement of crop quality in agricultural field," 2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018, vol. 2018-January, pp. 612–615, Feb. 2018, doi: 10.1109/CCWC.2018.8301662.

[35] H. Taherdoost, "A review of technology acceptance and adoption models and theories," Procedia Manuf, vol. 22, pp. 960–967, Jan. 2018, doi: 10.1016/J.PROMFG.2018.03.137.

[36] Y. W. Sek, H. Deng, E. McKay, and M. Qian, "Investigating the Impact of Learners Learning Styles on the Acceptance of Open Learner Models for Information Sharing," May 2016, Accessed: Apr. 17, 2025. [Online]. Available: https://arxiv.org/abs/1606.00747v1

[37] S. Y. Wee, "An Empirical Study of Learners' Acceptance of Open Learner Models in Malaysian Higher Education," RMIT University, 2016.

[38] S. Liao, J. C. Hong, M. H. Wen, Y. C. Pan, and Y. W. Wu, "Applying Technology Acceptance Model (TAM) to explore Users' Behavioral Intention to Adopt a Performance Assessment System for E-book Production," Eurasia Journal of Mathematics, Science and Technology Education, vol. 14, no. 10, p. em1601, Jul. 2018, doi: 10.29333/EJMSTE/93575.

[39] L. Liu and Q. Ma, "Perceived system performance," ACM SIGMIS Database: the DATABASE for Advances in Information Systems, vol. 37, pp. 51–59, Sep. 2006, doi: 10.1145/1161345.1161354.

[40] R. C. Tsaur and Y. H. Lin, "Exploring the Consumer Attitude of Building-Attached Photovoltaic Equipment Using Revised Technology Acceptance Model," Sustainability 2018, Vol. 10, Page 4177, vol. 10, no. 11, p. 4177, Nov. 2018, doi: 10.3390/SU10114177.

[41] Kung-teck, - Wong, R. B. Osman, P. S. Choo, - Goh, and M. K. Rahmat, "Understanding Student Teachers' Behavioural Intention to Use Technology: Technology Acceptance Model (TAM) Validation and Testing," International Journal of Instruction, vol. 6, no. 1, pp. 2432–2440, Dec. 2012, doi: 10.1016/J.COMPEDU.2011.06.008.

[42] E. Toros, G. Asiksoy, and L. Sürücü, "Refreshment students' perceived usefulness and attitudes towards using technology: a moderated mediation model," Humanities and Social Sciences Communications 2024 11:1, vol. 11, no. 1, pp. 1–10, Feb. 2024, doi: 10.1057/s41599-024-02839-3.

[43] W. U. H. Shah, G. Hao, R. Yasmeen, H. Yan, and Y. Qi, "Impact of agricultural technological innovation on total-factor agricultural water usage efficiency: Evidence from 31 Chinese Provinces," Agric Water Manag, vol. 299, p. 108905, Jun. 2024, doi: 10.1016/J.AGWAT.2024.108905.

[44] A. Bagheri and N. Emami, "Famers' Intention to Use Precision Farming Technologies, Application of the Extended Technology Acceptance Model: A Case in Ardabil Province," Journal of Agricultural Science and Technology, vol. 27, no. 1, pp. 81–95, 2025, doi: 10.22034/JAST.27.1.81.

[45] N. Wilson, K. Keni, and P. H. P. Tan, "The role of perceived usefulness and perceived ease-of-use toward satisfaction and trust which influence computer consumers' loyalty in China | Gadjah Mada International Journal of Business," Gadjah Mada International Journal of Business, 2021, Accessed: Apr. 17, 2025. [Online]. Available: https://search.informit.org/doi/epdf/10.3316/informit.147511565887487

[46] A. Z. Bayih, J. Morales, Y. Assabie, and R. A. de By, "Utilization of Internet of Things and Wireless Sensor Networks for Sustainable Smallholder Agriculture," Sensors 2022, Vol. 22, Page 3273, vol. 22, no. 9, p. 3273, Apr. 2022, doi: 10.3390/S22093273.

[47] M. McCaig, R. Dara, and D. Rezania, "Farmer-centric design thinking principles for smart farming technologies," Internet of Things, vol. 23, p. 100898, Oct. 2023, doi: 10.1016/J.IOT.2023.100898.

[48] I. A. C. Jimenez, L. C. C. García, F. Marcolin, M. G. Violante, and E. Vezzetti, "Validation of a TAM Extension in Agriculture: Exploring the Determinants of Acceptance of an e-Learning Platform," Applied Sciences 2021, Vol. 11, Page 4672, vol. 11, no. 10, p. 4672, May 2021, doi: 10.3390/APP11104672.

[49] J. Pfeiffer, A. Gabriel, and M. Gandorfer, "Understanding the public attitudinal acceptance of digital farming technologies: a nationwide survey in Germany," Agric Human Values, vol. 38, no. 1, pp. 107–128, Feb. 2021, doi: 10.1007/S10460-020-10145-2/TABLES/5.

[50] M. E. Marescotti, E. Demartini, R. Filippini, and A. Gaviglio, "Smart farming in mountain areas: Investigating livestock farmers' technophobia and technophilia and their perception of innovation," J Rural Stud, vol. 86, pp. 463–472, Aug. 2021, doi: 10.1016/J.JRURSTUD.2021.07.015.

[51] M. McCaig, D. Rezania, and R. Dara, "Is the Internet of Things a helpful employee? An exploratory study of discourses of Canadian farmers," Internet of Things, vol. 17, p. 100466, Mar. 2022, doi: 10.1016/J.IOT.2021.100466.

An Analytical Review of Environmental and Machine Learning Approaches in Dengue Prediction

Orlando Iparraguirre-Villanueva*, Juan Chavez-Perez, Eddier Flores-Idrugo, Luis Chauca-Huete
Facultad de Ingeniería, Universidad Tecnológica del Perú, Lima 15306, Perú

***Abstract*—In recent years, dengue has gained prominence as a priority public health challenge due to increasing incidences of spread. The main objective of this systematic literature review (SLR) is to explore the use of environmental factors and machine learning (ML) techniques to combat dengue, based on studies published between 2020 and 2024. For this purpose, 56 studies were selected from a balanced distribution of PubMed, Web of Science, Scopus and Springer Link, under the Preferred Reporting Items for Systematic Reviews and meta-analyses (PRISMA) method. The results obtained made it possible to determine that the climatological variables, such as temperature difference, humidity concentration and rainfall volume, are conditioning factors in the spread of the dengue virus. As for ML models, Random Forest and Support Vector Machines proved to be more accurate than traditional methods in detecting risk areas. The highest scientific production corresponded to the year 2024, with 25% of the studies, while India, with 14.29%, and the United States, with 12.50%, stood out as the countries with the highest contribution. In conclusion, ML techniques have enormous potential for strengthening early detection systems and optimizing resources in high-risk areas, but further research is needed in this field due to the lack of data availability and replicability of models.**

***Keywords*—Public health analytics; machine learning models; disease prediction; environmental risk factors; dengue surveillance; health data analysis**

I. Introduction

In recent decades, dengue, a viral disease transmitted by bites from mosquitoes, such as Aedes albopictus and Aedes aegypti, has gained prominence as a major public health threat, affecting both tropical and subtropical regions. The increasing prevalence of the disease, together with the challenges associated with its control in areas of high population density, underscores the need to address this problem on a global scale, a concern supported by the United Nations (UN), which has pointed out the effects of climate change on the geographical spread of dengue [1]. In this regard, the World Health Organization (WHO) has identified dengue as a global priority, promoting various strategies and policies for its control and prevention in endemic areas [2]. The scenario is conditioned by various factors that create conditions conducive to spread in the environment.

The proliferation of mosquito vectors is directly linked to climate, especially variations in temperature, rainfall and humidity, which facilitate the spread of dengue [3]. Abnormal climatic factors and rainfall create favorable conditions for the transmission of the virus, which increases the risk in both urban and rural areas, raising the incidence of the disease [4], [5]. Climate change has altered the natural habitats of these mosquitoes, favoring their expansion into new regions. In Spain, the presence of Aedes albopictus since 2014 has raised the risk of autochthonous transmission in Mediterranean areas [6]. In February 2023, two cases of dengue were reported in Ibiza, Spain, related to the presence of the transmitting mosquito, detected in the region since 2014. This fact highlights the importance of monitoring non-endemic areas where the vector has been introduced [7]. In 2018, an outbreak of dengue hemorrhagic fever in Vietnam underlined the importance of early detection by real-time reverse transcription (RT-PCR), a technique that made it possible to identify the serotypes responsible and prevent new outbreaks [8]. Pharmacological investigations have identified compounds such as CHEMBL376820, which exhibit inhibitory properties against dengue serotype 2, representing a significant advance in treatment strategies [9]. The use of ML artificial intelligence technique has shown a significant impact on the optimization of dengue prevention and control strategies, allowing foreseeing outbreaks and improving interventions in at-risk areas. In Espírito Santo, Brazil, Aedes albopictus was identified as a transmitter of DENV-1 and ZIKV, complicating control efforts in both urban and rural areas [3]. Regarding diagnostic techniques, triplex polymerase chain reaction with reverse transcription and real-time quantification (Triplex RT-qPCR) has been widely recognized for its high sensitivity and specificity in detecting serotypes in endemic areas [10]. ML-based predictive models, such as Gradient Boosting Machines (GBM) and Artificial Neural Networks (ANN), achieved an accuracy of 80.6% in identifying key factors, which improves dengue control decisions [11]. Furthermore, ANNs, within these models, have demonstrated high performance, with receiver operating characteristics of 0.8324 and an accuracy of 0.7523. The main risk factors identified include age, antigenemia on-structural protein 1 of the dengue virus (NS1) and the coexistence of Immunoglobulin M (IgM) and Immunoglobulin G (IgG). This approach could facilitate rapid prognoses during outbreaks, interpreting immunological dynamics and contributing to vaccine development [12].

*Corresponding author.

In terms of immunization, the Dengvaxia vaccine has been shown to be effective in children between 9 and 16 years of age who have previously experienced the disease, significantly reducing the number of severe cases and patients requiring hospitalization. However, before administering the vaccine, it is essential to perform blood tests to determine whether the individual has previously contracted dengue, as there is an increased risk of serious complications if the disease has not been experienced [13]. Similarly, the tetravalent mRNA vaccine has demonstrated promising immune responses, positioning itself as a viable alternative in the prevention of dengue virus [14].

The present study is justified by the growing global concern about the impact of dengue on public health, particularly in those regions where environmental factors facilitate its spread. Despite advances in control, the disease continues to represent a global challenge. This paper aims to review the literature to identify the main environmental factors that contribute to dengue transmission and to evaluate the potential of ML techniques to improve outbreak prediction, prevention and control strategies, to contribute to the implementation of more effective interventions.

II. Literature Review

Previous research on dengue has highlighted technological advances that have contributed to improved diagnostic accuracy and accessibility. The study [15] developed a microfluidic device (Cygnus) used for rapid detection and serotyping of NS1 protein in plasma samples using multiplex immunoassays, resulting in 82% sensitivity and 86% specificity, with potential for epidemiological surveillance. Similarly, the study [16] identified immunoreactive peptides corresponding to E and NS1 proteins to discover dengue-specific IgM and IgG antibodies, using peptide synthesis tests and ELISA, obtaining results with sensitivity levels of (73.33-96.66%) and specificity levels of (82.14-100%), offering an economically cost-effective solution for diagnosis by serology.

The work of [17] implemented a machine learning model to predict Aedes albopictus suitability on a global scale based on climatic and environmental data, which revealed that suitable habitats will expand markedly in the northern hemisphere, exposing an additional billion people by the mid-21st century. On the other hand, research by [18] examined the prevalence of DENV-1 and DENV-2 serotypes in Karachi, Pakistan, using immunochromatographic and PCR detection techniques, identifying that 80% of cases corresponded to DENV-2, associating its spread to favorable climatic conditions and disorganized urbanization. Finally, the analysis of [19] explored the impact of climate change on the distribution of Aedes albopictus in China, using data observed between 1970 and 2021 along with classification tree models, which allowed projecting that the risk of dengue will cover almost all of China by 2050 and 2080, affecting 1.2 billion people, with winter temperatures and summer precipitation as key factors in its spread.

The study by [20] developed and validated a triplex RT-qPCR protocol to detect chikungunya, dengue and zika viruses in mosquitoes simultaneously, employing specific primers and probes, demonstrating low detection thresholds ($1.32x10^0$ for CHIKV, $3.79x10^0$ for DENV1-4 and $2.06x10^0$ for ZIKV), so there was no cross-reactivity with homogenates of Aedes aegypti mosquitoes, evidencing its suitability for vector surveillance in endemic areas. In turn, the research of [21] evaluated the VECTRACK system, which combines optical sensors and machine learning, for automatic counting and classification of Aedes albopictus and Culex pipiens in field conditions. The results showed a high correlation between visual and automatic identification (Spearman: 0.97 for females and 0.89 for males), with an efficiency comparable to conventional traps, which supported its potential for continuous monitoring with minimal human intervention.

In [22], they addressed the prediction of dengue outbreaks in Bangladesh by means of climate data analysis and the application of machine learning algorithms, such as Support Vector Machine (SVM), Decision Tree and Random Forest, achieving an accuracy of 96.73% with the SVM model, confirming that temperature, humidity and rainfall are fundamental predictive factors. In the research by [23], they developed a methodology to predict dengue incidences in 20 Brazilian cities, integrating climatological data, Internet searches and autoregressive terms, where the Random Forest and LASSO regression models proved to be robust, with global search data as strong predictors. In addition, the study by [24] applied ensemble machine learning techniques to identify factors associated with dengue transmission, analyzing variables such as vapor pressure and precipitation, which improved the accuracy of the predictive models and highlighted the usefulness of ML in mosquito-borne disease surveillance.

It should be noted that numerous studies on the subject highlight the importance of adopting modeling and computational engineering techniques for the analysis of this type of disease. For example, the study [25] made use of ML practices known as logistic regression, ANN and bagging assembly, aimed at predicting the inherent risk of shock in patients carrying dengue virus, using clinical and physiological data collected at a university-level medical facility, highlighting that the bagging assembly method achieved a 14.5% increase in accuracy compared to other approaches, identifying hemoglobin as a key indicator for predicting severe complications. Also, in [26], they proposed a fractional mathematical derivation to analyze the dynamics of Zika and dengue co-infection, validating the model with epidemiological data from Brazil and sensitivity analysis. The results showed that the rate of human transmission of Zika and vector mortality are determining factors in controlling both diseases. Table I presents a summary of the main findings and limitations of the literature review.

TABLE I. SYNTHESIS OF FINDINGS AND LIMITATIONS IN DENGUE LITERATURE

References	Main Resulted	Limitations
[15]	Cygnus device detects NS1 with 82% sensitivity in 40 minutes in multiplex.	Requires enhancement of antibodies to DENV-1. Do not test in rural areas.
[16]	E/NS1 protein peptides identify IgM/IgG (73.33-96.66%) in sensitivity.	Cross-reactivity with other Flaviviruses. Cost not evaluated.
[17]	ML model predicts global expansion of A. albopictus in the northern hemisphere.	Omits vector control policies in projections.
[18]	DENV-2 predominant (80%) in Karachi, associated with disorganized urbanization.	Limited to one Pakistani city. Does not analyze future climate.
[19]	Winter temperature key to A. albopictus distribution in China between 93.0% to 98.8% accuracy.	Does not validate with socioeconomic data.
[20]	Triplex RT-qPCR detects CHIKV/DENV/ZIKV in mosquitoes, with no cross-reactivity.	Detection thresholds are not compared to other methods.
[21]	VECTRACK system identifies Aedes/Culex with 97% correlation in females and 89% correlation in males with visual-ML.	Only tested in Italy. Does not evaluate cost-effectiveness.
[22]	SVM predicts dengue outbreaks with 96.73% accuracy in temperature and rainfall climates.	Data limited to Bangladesh.
[23]	Random Forest and LASSO predict incidence in 20 Brazilian cities.	Requires real-time internet search data.
[24]	Ensemble models improve accuracy in dengue prediction with vapor pressure and precipitation.	Does not specify omitted climatic variables.
[25]	Bagging algorithm predicts dengue shock with 14.5% more accuracy than hemoglobin indication.	Data from a single medical center.
[26]	Fractional mathematical model validates Zika-Dengue co-dynamics in Brazil.	Does not include public health interventions.

III. METHODOLOGY

This section will describe the procedures used to carry out the systematic study on models of ML and environmental variability applicable to the prediction of dengue outbreaks, in accordance with PRISMA 2020 regulations.

A. Purpose and Research Questions

The study comprehensively analyzed scientific literature on the integration of environmental variables such as climatic, ecological and urban variables with ML models for dengue outbreak forecasting. By means of the PRISMA 2020 method, the latest trends, challenges and opportunities for improvement in predictive models were identified, with the purpose of guiding future research towards solutions of greater precision and relevance in the field of public health. In this sense, the following central question arises as the main objective of the analysis:

Question 1: What are the advances and limitations identified in the scientific literature on the use of environmental factors and ML techniques in dengue prediction?

B. Sources of Information and Search Strategy

The methodical search for the studies was carried out on February 7, 2025; for this purpose, four renowned databases in the field of health sciences and cutting-edge technologies were consulted: PubMed, Web of Science, Scopus, and Springer Link. The criteria for selecting these sources were their capacity for exhaustive indexing of peer-reviewed literature, as well as their editorial impact factor and their thematic relevance according to the objectives of the study.

In the development of the search strategy, standardized descriptors combined with Boolean operators were used. The unified conceptual structure was as follows: ("Public Health Analytics" OR "Health Data Analysis") AND ("Machine Learning Models" OR "Deep Learning Algorithms") AND ("Disease Prediction" OR "Epidemic Forecasting") AND ("Environmental Risk Factors" OR "Climatic Determinants" OR "Urban Health Indicators") AND ("Dengue Surveillance" OR "Arbovirus Monitoring").

The search code was adapted to the syntactic and functional characteristics of each database. Despite the structural adaptations, the integrity of the search was preserved in all cases.

In addition, temporal filters were performed to restrict the search exclusively to documents published between 2020 and 2024, as well as language filters, so that only documents written entirely in English were included. Furthermore, using databases that allowed this, including PubMed, an additional restriction was introduced to limit the availability of text (abstract, full text) and species (human).

As a result of the systematic search process, a total of 1593 documents were obtained. Fig. 1 shows their distribution by database source and provides a quantitative view of the coverage and retrieval efficiency.

C. Inclusion and Exclusion Criteria

Research was selected by prioritizing climatic conditions, epidemiological patterns and predictive approaches. Inclusion and exclusion criteria applicable in this review of the scientific literature are detailed below.

1) Inclusion criteria

a) Research was published between 2020 and 2024, to ensure that the evidence collected reflects the most recent advances in the field.

b) Studies that provide primary data on the use of ML techniques or the impact of environmental factors on dengue prediction and control.

c) Documentation available in English only, to maintain consistency and facilitate interpretation of results.

d) Papers that explore aspects such as predictive models, climatic variables, vector ecology or dengue transmission dynamics related to the focus of study relevant to the research question

2) Exclusion criteria

a) Publications outside the 2020–2024-time range, since they would not reflect the current state of knowledge.

b) Articles that do not provide empirical evidence, such as reviews, meta-analyses, commentaries or theoretical studies without original data.

c) Papers in languages other than English, to avoid inconsistencies in interpretation and analysis.

d) Research that is not directly related to dengue, its environmental factors or ML-based prediction methods.

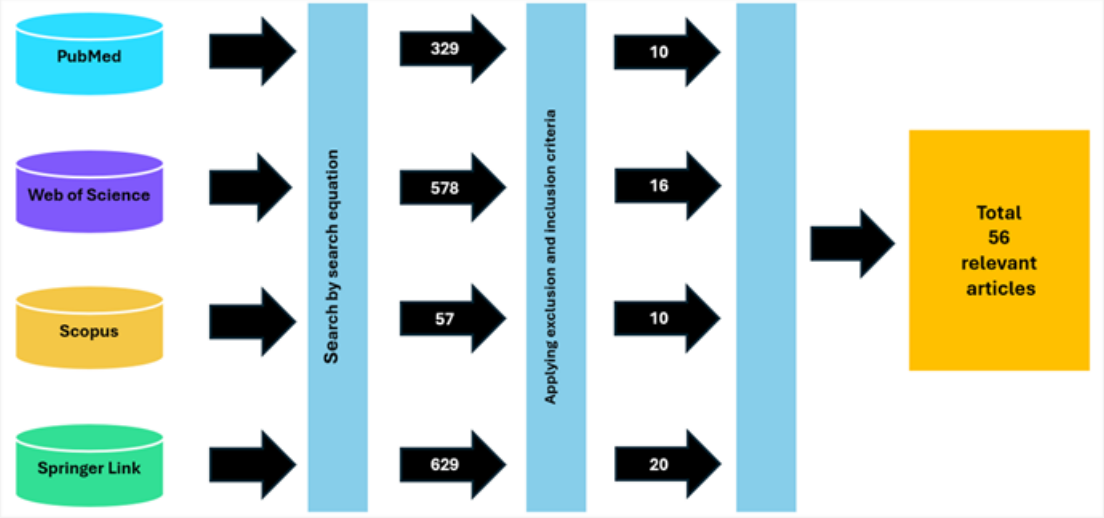


Fig. 1. Distribution of studies by database.

D. Study and Data Selection Process

The records were managed using the Mendeley® reference manager, where specific collections were created for each database. Subsequently, the records were exported and imported in ".RIS" format to Rayyan®, a tool specialized in systematic reviews, allowing efficient detection and categorization of duplicates.

After the screening phase, the system automatically detected and eliminated 222 duplicate papers. In addition, 32 other secondary literature articles, including systematic reviews and meta-analyses, were excluded according to pre-established eligibility criteria.

The completeness of the selection process was carried out jointly by three reviewers who worked in consensus from the beginning in rigorous analysis procedures to minimize bias in each case; however, if difficulties or disagreements arose regarding the inclusion of certain literary studies, a fourth reviewer, a specialist in methodology, intervened to resolve those discrepancies to ensure methodological consistency. In this way, we were able to select only the studies most relevant to the focus of the study, guaranteeing the rigor and transparency of the review. Table II shows the number of duplicates and systematic reviews eliminated by database source.

After the selection phase in the Rayyan environment, we proceeded to work with the eligible studies to organize them in a data matrix established in Microsoft Excel spreadsheets designed specifically for the systematized analysis. The matrix included previously defined categorization fields to disaggregate the information, such as the database of origin, the title of the article, the year of publication, the type of publication, the country of origin, the methodological approach (qualitative, quantitative or mixed), and the results directly related to the research objective.

TABLE II. DISTRIBUTION OF DUPLICATES AND SYSTEMATIC REVIEWS BY DATABASE

Database	Duplicates	Systematic Reviews
PubMed	16	13
Web of Science	123	0
Scopus	3	2
Springer Link	80	17
Total	**222**	**32**

The import of data grouped by bibliographic source of origin was carried out by two reviewers of the team, who worked closely together to ensure accuracy and completeness. A third member of the team inspected and verified the existence of inconsistencies or possible misinterpretations within the data. At this stage, it was not necessary to contact the original authors of the studies, since the information contained in the articles allowed the analysis to be carried out.

The work methodology, characterized by its structure and collaborative access developed by the group, made it possible to identify thematic patterns and methodological trends, particularly in the integration of environmental variables in dengue transmission prediction models. Fig. 2 shows the distribution of the documents, showing the number of items collected, discarded and finally selected using the PRISMA 2020 Shiny tool.

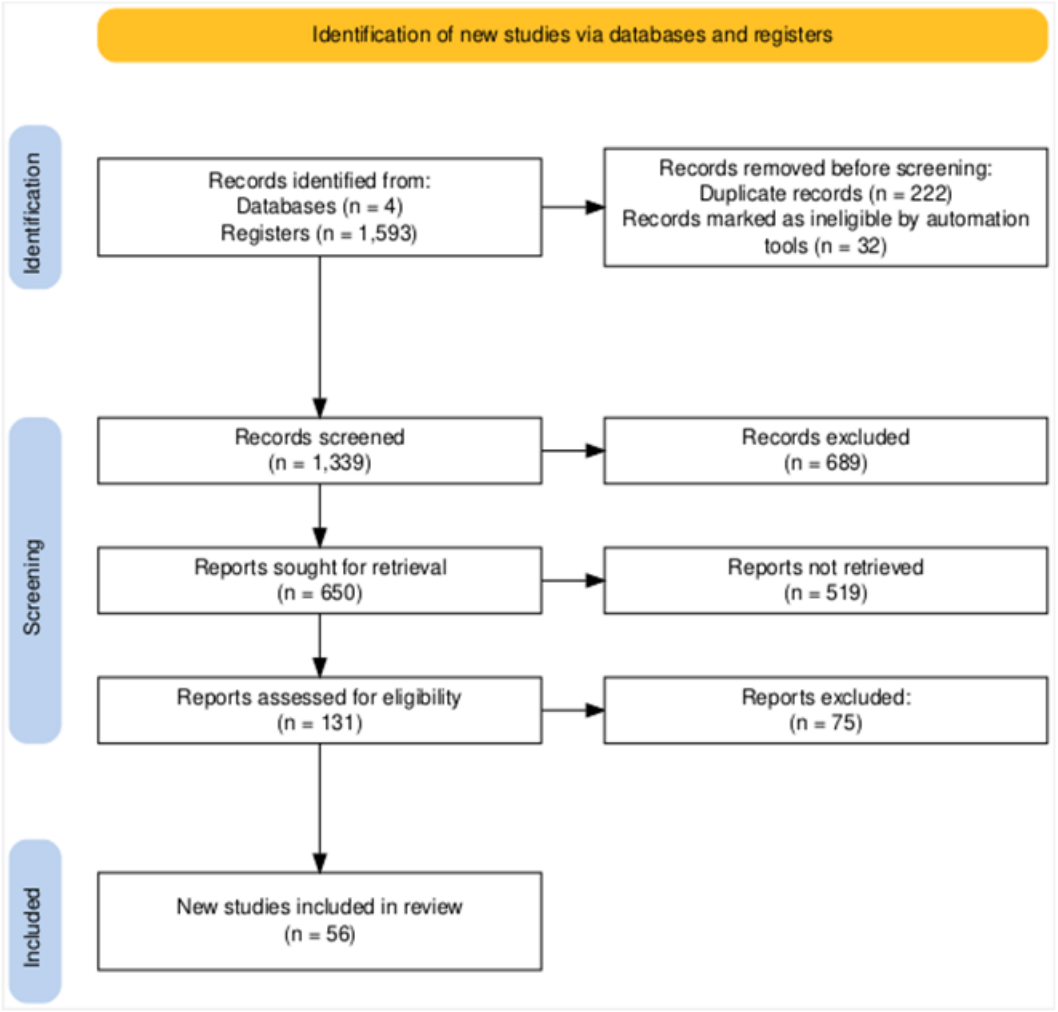


Fig. 2. PRISMA method.

E. Type of Study

The PRISMA methodology was implemented in this study according to the guidelines up to version 2020, to ensure systematic rigor [27]. Regarding the elaboration of the PRISMA flowchart, an R package with the specific application Shiny was used, which enhanced transparency through the graphical representation of the selection phases [28]. A strict application of eligibility criteria and detailed documentation contributed to strengthening validity and replicability.

F. Risk of Bias Assessment

This review did not use standardized instruments to assess the risk of bias in the primary studies. This decision responds to the descriptive and exploratory nature of the analysis, which did not include quantitative syntheses or meta-analyses that required weighing individual methodological quality. However, as an alternative measure of rigor, we applied rigorous inclusion criteria that excluded secondary literature, duplicate studies, and publications without direct empirical evidence. This ensured that only primary research relevant to the qualitative analysis was incorporated into the discussion, while critically examining the methodological limitations of the studies included in the review. PRISMA checklist 2020 is given in Appendix (Table IX).

IV. Results

The results of this SLR were structured under the methodological standard defined by PRISMA. Initially, a total of 1593 articles were identified from the PubMed, Web of Science, Scopus and Springer Link databases. Achieving this initial compilation was the result of using the predefined keywords and their combinations in search strings in a rigorous manner. Table III details the distribution of the articles collected in this phase, including the number and percentage corresponding to each database.

TABLE III. Distribution of Studies Initially

Database	Quantity	Percentage
PubMed	329	20.65%
Web of Science	578	36.28%
Scopus	57	3.58%
Springer Link	629	39.49%

After numerically counting, the collected documents, a preliminary analysis was performed, which led to the creation and application of initial filters in Rayyan, including the elimination of duplicates and the exclusion of non-relevant documents. This process was critical, as it allowed the exclusion of non-primary documents, which ensured the quality of the documents. The Rayyan system detected duplicates, and after manually evaluating the articles placed under observation, 254 articles were excluded (222 duplicates and 32 systematic reviews), thus reducing the set of studies to 1339. This distribution by origin, quantity, and percentage value is shown in Table IV.

TABLE IV. DISTRIBUTION OF STUDIES IN PHASE 1

Database	Quantity	Percentage
PubMed	300	22.40%
Web of Science	455	33.98%
Scopus	52	3.88%
Springer Link	532	39.73%

Subsequently, in "Phase 1", relevant filters were applied to visually analyze the titles and specific keywords related to dengue and its vectors, such as Aedes albopictus, Aedes aegypti, Dengue virus and Mosquito Vectors. Performing this filtering allowed discarding 689 studies, since they were not aligned with what the research seeks, due to that the set was reduced to 650 studies. Table V shows the distribution of the studies selected in this phase, evidencing the rigorous application of the thematic relevance criteria.

TABLE V. DISTRIBUTION OF ITEMS IN PHASE 1

Database	Quantity	Percentage
PubMed	183	28.15%
Web of Science	209	32.15%
Scopus	27	4.15%
Springer	231	35.54%

In Phase 2, a hybrid process combining automated filters and manual review of the abstracts of the remaining 650 articles was carried out. This approach made it possible to assess the relevance of each study in terms of its objective, methodology and results, applying predefined inclusion and exclusion criteria. For the process, we used words more focused on what we sought to analyze in the abstracts of each study, the inclusion words used were "Aedes albopictus", "Dengue virus", "Public Health Analytics", "Disease Prediction", "Machine Learning Models" and "Environmental Risk Factors". Likewise, we sought to exclude studies that included topics such as "cells", "prevalence", "in vitro", "animal models" and "mice", since they were not aligned with the objectives of the review. A total of 519 studies were excluded, and the selection set was reduced to 131, as shown in Table VI.

TABLE VI. DISTRIBUTION OF ARTICLES IN PHASE 2 AFTER REVIEW OF ABSTRACTS

Database	Quantity	Percentage
PubMed	41	31.30%
Web of Science	22	16.79%
Scopus	12	9.16%
Springer Link	56	42.75%

Finally, for the last process called "Phase 3", the 131 studies selected up to the previous phase were reviewed in full text. This process involved the exhaustive and detailed evaluation of each study to determine whether its content information responded to the research question and met the established quality and relevance criteria. As a result, 75 additional articles were excluded, resulting in a final set of "56" studies of which 55 corresponded to journal articles and 1 to conference proceedings. Table VII presents the final distribution of the included studies that fully complied with the SLR requirements, showing their origin, quantity and percentage value of the total.

TABLE VII. FINAL DISTRIBUTION OF ITEMS IN PHASE 3

Database	Quantity	Percentage
PubMed	10	17.86%
Web of Science	16	28.57%
Scopus	10	17.86%
Springer Link	20	35.71%

As mentioned above, the completeness of the study selection process, from the initial count to the final phase 3, was carried out entirely in Rayyan software, supported by data in Microsoft Excel, guaranteeing transparency and thoroughness at each stage. Fig. 3 shows the final distribution of the 56 selected studies, classified according to the databases used for this SLR, while the length is reflected in the bar graph corresponding to the exact number of studies included by each source, highlighting the contribution of each one to the final set.

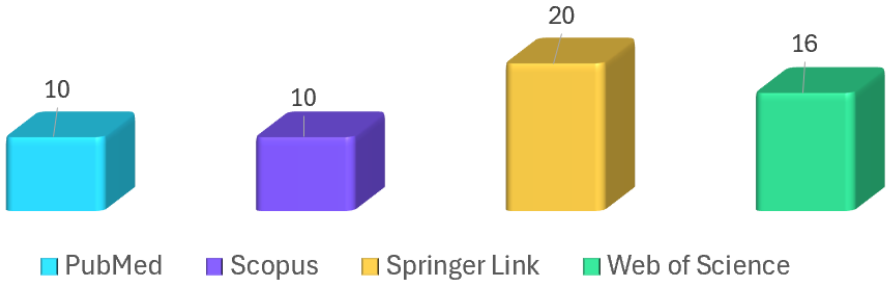


Fig. 3. Distribution of studies by databases.

The analysis of the temporal distribution by number of studies pertaining to the years between 2020 and 2024 reveals significant concentrations of impact. The year 2024 stood out as the year with the highest production, representing 25% of the total number of publications, with 14 documents. This was followed by the year 2021, which contributed 13 with a value of 23.21%, while 2020 contributed 12 documents at 21.43%. For their part, 2023 and 2022 presented a lower contribution, with 11 in 19.64% and 6 in 10.71%, respectively. The trend suggests that the topic has been addressed by researchers over time, highlighting the increase in ML techniques, which in turn consider environmental factors to be important in combating dengue in recent years. Fig. 4 illustrates this temporal distribution, highlighting the relevance of the studies published in the most recent period.

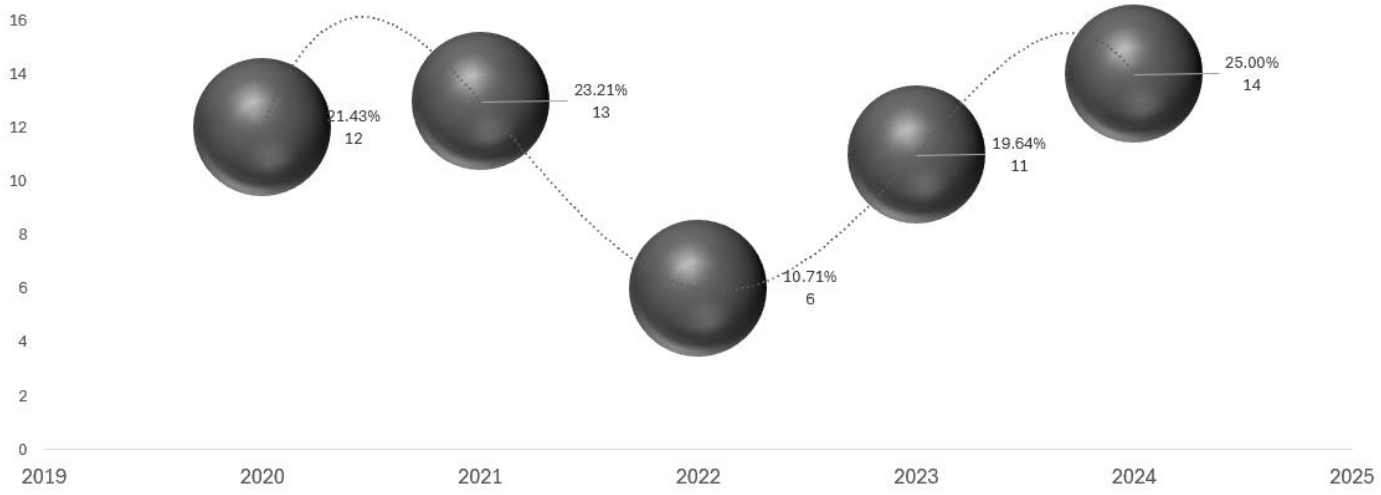
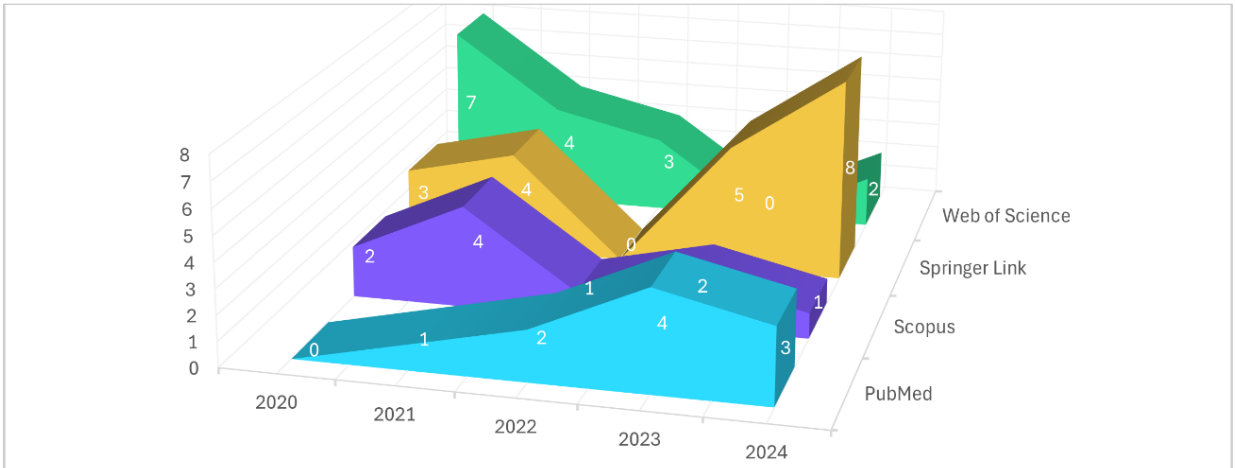


Fig. 4. Number of studies by year of publication.

When the origin of the studies was broken down according to the databases and their annual distribution, interesting patterns were observed. In 2020, most studies came from Web of Science with a value of 7, followed by Springer Link with 3 and Scopus at 2, while PubMed recorded no contributions. Regarding 2021, contributions were more evenly distributed, where Scopus, Springer Link and Web of Science had several 4 each respectively, and only PubMed had 1. For 2022, Web of Science again stood out with 3, while PubMed contributed 2 and Scopus 1, with no contributions from Springer Link. In 2023, Springer Link registered the highest contribution with 5 articles, followed by PubMed with 4 and Scopus with 2, with no studies coming from Web of Science. Finally, in 2024, Springer Link led again with 8 studies being the one with the highest impact by contribution, followed by PubMed with 3, Scopus with 1 and Web of Science with 2. Fig. 5 visually presents evolutionary distribution, highlighting the periods analyzed.



	2020	2021	2022	2023	2024
PubMed	0	1	2	4	3
Scopus	2	4	1	2	1
Springer Link	3	4	0	5	8
Web of Science	7	4	3	0	2

Fig. 5. Number of studies per year and databases.

The geographical distribution of the 56 studies included reflects the participation of several countries among which the most relevant belong to the continental regions of Asia and North America. In this sense, India was the main contributor, with 8 articles worth 14.29% of the total, followed by China with 6 articles worth 10.71% and the United States with 7 articles worth 12.50%. On the other hand, in Europe, Italy stood out with 4 studies at 7.14%, while countries such as France, Spain and Switzerland made more modest contributions. In other regions, Malaysia with 5 at 8.93% and Bangladesh with 4 at 7.14% also showed significant participation. On the other hand, countries such as Ghana, Pakistan, Tanzania and Mexico, among others, contributed 1 to 3 studies each, representing smaller percentages. This geographical distribution is evidence of the globalization of research on the subject, although with a notable concentration in countries with greater scientific and technological development. It should be noted that all the papers were published in English, which reinforces its role as the predominant language in global scientific dissemination. Fig. 6 presents this geographical distribution on a global scale in detail.



Fig. 6. Geographical distribution of publications.

The studies included were classified according to their type of methodological approach, obtaining results where the quantitative approach was the most representative with a value of 98.21%, while it was identified that there was only one study with the mixed approach that composed a qualitative and quantitative analysis and therefore obtained its mixed label in 1.79%. Regarding the membership of Springer Link databases, it contributed the most quantitative studies represented in 19 with 33.93%, followed by Web of Science in 16 with 28.57%, Scopus and PubMed with 10 in 17.86% respectively. The only study with a mixed approach came from Springer Link. This methodological distribution reflects the preference for quantitative approaches in dengue-related research, possibly due to the nature of the data and the ML techniques applied.

As for the type of research, all 56 included studies correspond to completely different distributions due to the quantities, where there are original articles with 98.21% and conference proceedings with 1.79%. The predominance of original articles highlights the relevance of this format in the generation of knowledge on the application of ML techniques and environmental factors in dengue control.

To deepen the thematic analysis of the selected studies, the VOSviewer tool was used to construct a network of co-occurrences of keywords. It was possible to observe how research on dengue and its vectors is organized into thematic clusters clearly differentiated by color. The classification by clusters allows an understanding of the direct relationships between keywords, establishing specific themes, where the red cluster represents the attraction and repellency of mosquitoes by certain stimuli colored odors or CO2, whose definition is associated with terms such as "Aedes aegypti", "repellent" and "carbon-dioxide". On the other hand, the green cluster, constituted by connections such as "larvicidal", "Culicidae" and "plant-extracts", exposes the strategies of larval control and the understanding of their interaction with ecology. In the blue cluster, such as "prevalence", "dengue fever" and "optimal control" for efforts to understand how the disease spreads and how to control it effectively. Meanwhile, the purple cluster in "public health" and "citizen science", which emphasizes the importance of community involvement and public health policies in preventing outbreaks.

Finally, it is highlighted that the strongest connections in the co-occurrence network is that between "Aedes albopictus" belonging to the pink cluster and "dengue" to the purple cluster, visually it can be seen that both words have a considerable size with respect to the others, since their size represents the highest number of mentions in research, confirming the relevance of this mosquito in the transmission of the disease. Fig. 7 visually illustrates these relationships, offering a clear and proportional overview of the most researched topics and their connections.

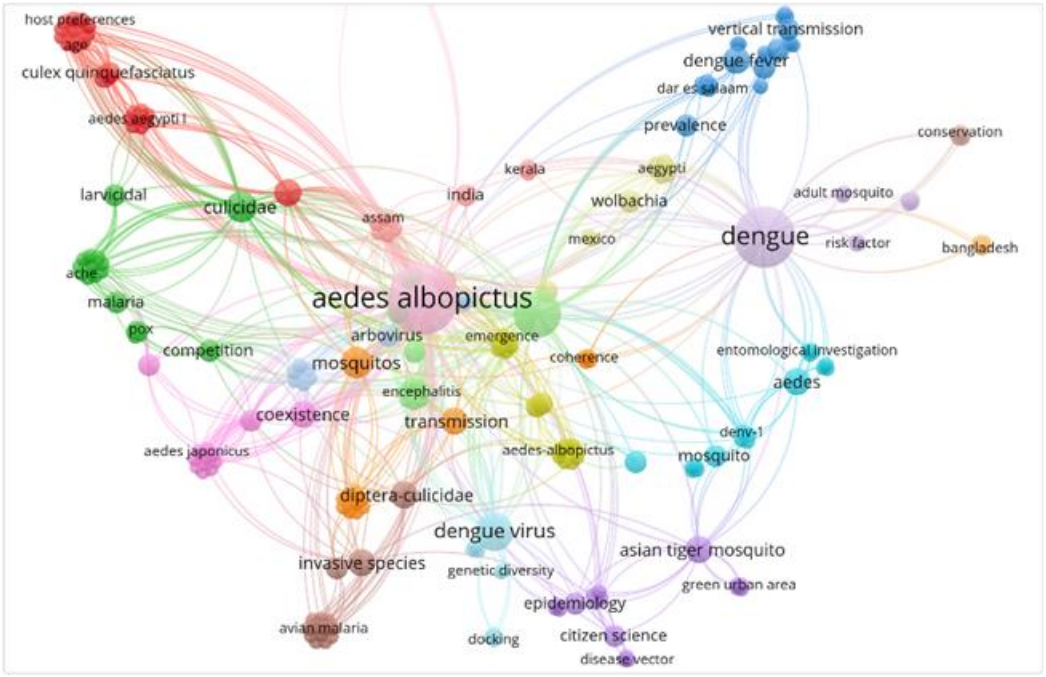


Fig. 7. Bibliometric scan of literature.

V. Discussion

A. What are the Advances and Limitations Identified in the Scientific Literature on the Use of Environmental Factors and ML Techniques in Dengue Prediction?

The possibilities of predicting the occurrence of dengue through environmental factors and ML techniques constitute a field of opportunities to address public health challenges. Recent studies in scientific literature highlight the disruptive potential of technology to achieve substantial progress in short periods of time. In this line, the application of ANNs stands out because they provide a high degree of accuracy when analyzing climatological variables such as temperature, humidity and precipitation. The combination of these techniques with the Mosquito Oviposition Index makes it possible to identify areas at risk with greater accuracy. However, there are limitations such as accessibility to historical data and overfitting of local value sets, which compromises the effectiveness of the predictions.

The work done by [22] used machine learning algorithms, such as SVM, Decision Tree and Random Forest, from which climate parameters observed in Bangladesh were analyzed, achieving an accuracy of 96.73%. This method scientifically demonstrated that climatic variables are a decisive indicator of the occurrence of epidemic outbreaks. On the other hand, in [23] a methodological framework was devised to forecast the occurrence of dengue cases in 20 Brazilian cities, by integrating climatic data, Internet queries and autoregressive terms. The robustness of the Random Forest and LASSO regression models was demonstrated, highlighting the potential of Internet browsing data as powerful predictive indicators.

In the domain of environmental factors, the study of [17] implemented machine learning models with the objective of predicting the global-scale suitability of natural habitats for Aedes albopictus as a direct function of climate and environmental data, showing results that indicate that mosquito-friendly habitats will expand markedly in the northern hemisphere, exposing a billion people by the middle of the 21st century. Similarly, research [19] analyzed the impact of global warming on the distribution region of Aedes albopictus in China, using data observed during the period from 1970 to 2021 combined with classification tree models, the result of which projected that the risk of dengue would cover China in its entirety in future years between 2050 and 2080, reaching 1.2 billion people affected, due to winter temperatures and summer precipitation as the main factors in its propagation.

In [25], they developed a ML-based predictive model for the statistical calculation of the risk of shock in patients with dengue, using clinical and physiological data collected in a university medical center, being combined models such as logistic regression and ANN, achieving that the bag assembly method obtained a 14.5% increase in accuracy compared to other approaches, identifying hemoglobin as a fundamental predictor of the severity of complications. In this regard, the study [26] provided a fractional derivative mathematical model for the analysis of Zika and dengue codynamics. The model was subjected to validation tests based on epidemiological data from Brazil and sensitization analyses, the results of which showed that the speed of human transmission and vector mortality are conditioning factors for the control of both diseases, suggesting strategies aimed at attenuating human-to-human transmission.

Among the limitations identified, overfitting to local data, noted in [22], reduces the ability of the models to generalize to other regions. In addition, the lack of complete historical climate data, mentioned in [17], limits the accuracy of

predictions based on environmental factors. Furthermore, another considerable challenge is the high cost of implementing advanced technologies, such as the VECTRACK system evaluated in [21], whose effectiveness requires considerable investments in infrastructure and training. Progress in this area is presented in Table VIII, where the advances and limitations found in literature are grouped as follows.

TABLE VIII. ADVANCES AND LIMITATIONS IN DENGUE PREDICTION WITH ENVIRONMENTAL FACTORS AND ML

#	Advances	Limitations	Quantity	References
1	Outbreak prediction using temperature and Mosquito Oviposition Index (MOI)	Incomplete climate data; low spatial and temporal resolution	6	[29], [30], [31], [32], [33], [34]
2	ANN to predict incidence	Overfitting local data; limited generalizability; low interpretability	8	[35], [36], [37], [38], [39], [40], [41], [42]
3	Satellite data for mapping risk zones	High cost; coverage and cloudiness issues; high computational demand	5	[43], [44], [45], [46], [47]
4	Vertical transmission in mosquitoes	Poor generalization; limited epidemiological evidence	4	[48], [49], [50], [51]
5	Identification of antivirals with molecular docking	Limited experimental validation; absence of clinical trials	3	[52], [53], [54]
6	Network analysis to identify vectors	Dependence on quality data; risk of bias in incomplete data	4	[55], [56], [57], [58]
7	One Health" approach to surveillance	Complexity in interdisciplinary coordination; insufficient infrastructure	3	[59], [60], [61]
8	Vaccine development with immunogenic peptides	Low stability; limited immunogenicity; production barriers	4	[62], [63], [64], [65]
9	Vector control with Wolbachia	Difficulty of large-scale implementation; uncertainty regarding social acceptance	3	[66], [67], [68]
10	Prediction with socioeconomic factors	Incomplete socioeconomic data; regional heterogeneity	4	[69], [70], [71], [72]
11	Autoregressive Integrated Moving Average (ARIMA) models for time series	Limitations in long-term forecasts; low robustness to abrupt changes	3	[73], [74], [75]
12	Human mobility analysis	Restricted access to data; representativeness biases	3	[76], [77], [78]
13	Vector susceptibility by serotype	High variability between species and serotypes; lack of longitudinal data	4	[79], [80], [81], [82]
14	Deep Learning (DL) for outbreak prediction	Requires large volumes of data; risk of overfitting; low interpretability	2	[83], [84]

The results presented in Table VII show the various advances and limitations that exist in the development of environmental factors and ML techniques for dengue prediction. In this regard, the use of satellite data for mapping risk zones is relevant, allowing the delimitation of vulnerable areas with high precision. On the other hand, the focus on vertical transmission in mosquitoes offers valuable knowledge on the conservation of the vector in different populations. Regarding antiviral identification initiatives, certain techniques, such as molecular docking, show promising results, while network analysis has emerged as an innovative way to detect transmission patterns of the disease and its vectors. In terms of integrated surveillance systems for dengue viruses, the "One Health" approach brings together human health, animal health and the environment.

Despite these advances, many limitations remain. Among them, the high cost of implementing technologies such as satellite data and Wolbachia vector control makes them difficult to adopt in resource-poor regions. In addition, the limited experimental validation of antiviral compounds and the reliance on quality data in network analysis represent major challenges. At the same time, the complexity of interdisciplinary coordination of the "One Health" approach and limitations in large-scale production of peptide-based immunogenic vaccines highlight the need for improved cooperation and more infrastructure. Finally, the lack of accurate socioeconomic data and the difficulties in obtaining data on human mobility diminish the effectiveness of predictive models.

Likewise, a critical aspect identified in this review is the validation of ML models. Although it is true that several studies report significant metrics, it is important to exercise caution.

For example, the problem of overfitting in ANN and DL models can be observed, where local patterns are adjusted excessively. This limits their ability to generalize in contexts where geography is changing. Similarly, the problem of data leakage can also be observed, which occurs when the validation set is leaked into the training, and this can artificially inflate performance indicators.

Another important and well-known challenge is class imbalance. In many contexts, dengue outbreaks are a small fraction of the data compared to non-outbreak periods.

Consequently, it is recommended that future work address the following issues:

- Cross-validation that captures the sequential nature of outbreaks.
- Strategies to mitigate class imbalance.
- External generalization evaluations, testing models on data from different regions.

VI. Conclusions

This study addressed the relationship between environmental factors and the application of ML techniques in the prediction and control of dengue spread. For this purpose, an SLR was performed that analyzed 56 studies selected from various databases such as Springer Link which contributed 35.71% of the studies, followed by Web of Science with 28.57%, PubMed with 17.86% and Scopus with 17.86%, reflecting a representative and balanced distribution of the available scientific knowledge related to the subject during the period between 2020 and 2024.

Thanks to this research, it has been possible to discover essential trends linked to environmental factors that affect the spread of dengue, specifically the variation in temperature, humidity rate and precipitation regime. The effectiveness of various ML applications was also tested, highlighting the superiority of models based on supervised learning, such as Random Forest and SVM, for the classification and estimation of outbreaks compared to traditional approaches. On the other hand, it was found that the integration of geospatial and epidemiological data in the analyzed models is becoming more and more frequent, thus contributing to strengthening the accuracy in the identification of risk areas and planning of interventions.

This work will make a major contribution to public health research expertise by laying a strong foundation for the development of more potent anti-dengue measures. In particular, the findings emphasize the potential of forecasting methods to optimize early warning systems and focus resources on high-risk sectors. But there are also considerable limitations, such as the heterogeneity in the quality of the available data, which affects the comparability of the studies; the non-standardization of methodological approaches, which hinders the replicability of the models. These limitations are an opportunity for future research to optimize the integration of environmental data, improve the robustness of predictive models in real scenarios and explore innovative approaches, such as the use of unsupervised learning techniques to identify incipient patterns in the spread of dengue. In this way, it will be feasible to progress towards approaches that are more functional, durable and oriented to the specific needs of the territorial entities concerned by the disease.

Authors' Contributions

Conceptualization, O.I-V.; methodology, J.C-P, E.F-I and L.C-H; validation, J.C-P; formal analysis, J.C-P, E.F-I and L.C-H.; investigation, O.I.-V and J.C-P; data curation, E.F-I; writing—original draft preparation, O.I-V, J.C-P, E.F-I and L.C-H; writing—review and editing, O.I.-V; visualization, O.I-V, J.C-P, E.F-I and L.C-H. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable.

Data Availability Statement

The data used in this systematic review were collected from publicly accessible sources cited in the references. No primary data was generated.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] A. Mercier et al., "Impact of temperature on dengue and chikungunya transmission by the mosquito Aedes albopictus," Sci Rep, vol. 12, no. 1, Dec. 2022, doi: 10.1038/s41598-022-10977-4.

[2] L. Labes, N. C. T. Bordignon, and R. Gondak, "Synergism between dengue and COVID-19: evidence or occasional?," J. Oral Diagn, vol. 6, pp. e20210013–e20210013, Jan. 2021, doi: 10.5935/2525-5711.20210013.

[3] H. R. Rezende et al., "First report of Aedes albopictus infected by Dengue and Zika virus in a rural outbreak in Brazil," PLoS One, vol. 15, no. 3, 2020, doi: 10.1371/JOURNAL.PONE.0229847.

[4] Y. Zhang, L. Wang, G. Wang, J. Xu, and T. Zhang, "An ecological assessment of the potential pandemic threat of Dengue Virus in Zhejiang province of China," BMC Infect Dis, vol. 23, no. 1, Dec. 2023, doi: 10.1186/S12879-023-08444-0.

[5] B. Goh, P. Visendi, A. R. Lord, S. Ciocchetta, W. Liu, and M. T. Sikulu-Lord, "First Report of the Detection of DENV1 in Human Blood Plasma with Near-Infrared Spectroscopy," Viruses, vol. 14, no. 10, Oct. 2022, doi: 10.3390/V14102248.

[6] B. Fernández-Martínez et al., "Spatial analysis for risk assessment of dengue in Spain," Enfermedades infecciosas y microbiologia clinica (English ed.), vol. 42, no. 8, Oct. 2024, doi: 10.1016/J.EIMCE.2023.06.010.

[7] L. García-San-Miguel et al., "Detection of dengue in German tourists returning from Ibiza, Spain, related to an autochthonous outbreak, August to October 2022," Euro Surveill, vol. 29, no. 14, Apr. 2024, doi: 10.2807/1560-7917.ES.2024.29.14.2300296.

[8] H. T. ; Pham et al., "The Distribution of Dengue Virus Serotype in Quang Nam Province (Vietnam) during the Outbreak The Distribution of Dengue Virus Serotype in Quang Nam Province (Vietnam) during the Outbreak in 2018," Int. J. Environ. Res. Public Health, vol. 19, p. 1285, 2018, doi: 10.3390/ijerph19031285.

[9] S. K. Halder et al., "A Comprehensive Study to Unleash the Putative Inhibitors of Serotype2 of Dengue Virus: Insights from an In Silico Structure-Based Drug Discovery," Mol Biotechnol, vol. 66, no. 4, pp. 612–625, Apr. 2024, doi: 10.1007/S12033-022-00582-1.

[10] D. R. Kinanti et al., "Evaluation of in-house dengue real-time PCR assays in West Java, Indonesia," PeerJ, vol. 12, no. 7, 2024, doi: 10.7717/PEERJ.17758.

[11] B. C. Daniels et al., "Predicting the infecting dengue serotype from antibody titre data using machine learning," PLoS Comput Biol, vol. 20, no. 12, Dec. 2024, doi: 10.1371/JOURNAL.PCBI.1012188.

[12] S. W. Huang, H. P. Tsai, S. J. Hung, W. C. Ko, and J. R. Wang, "Assessing the risk of dengue severity using demographic information and laboratory test results with machine learning," PLoS Negl Trop Dis, vol. 14, no. 12, pp. 1–19, Dec. 2020, doi: 10.1371/JOURNAL.PNTD.0008960.

[13] G. Paz-Bailey et al., "Dengue Vaccine: Recommendations of the Advisory Committee on Immunization Practices, United States, 2021," MMWR Recomm Rep, vol. 70, no. 6, pp. 1–16, 2021, doi: 10.15585/MMWR.RR7006A1.

[14] M. Mukhtar, A. W. Wajeeha, N. Us Sahar, S. Zaidi, and N. Bibi, "Engineering Modified mRNA-Based Vaccine against Dengue Virus Using Computational and Reverse Vaccinology Approaches," 2022, doi: 10.3390/ijms232213911.

[15] S. H. Needs et al., "Smartphone multiplex microcapillary diagnostics using Cygnus: Development and evaluation of rapid serotype-specific NS1 detection with dengue patient samples," PLoS Negl Trop Dis, vol. 16, no. 4, p. e0010266, Apr. 2022, doi: 10.1371/journal.pntd.0010266.

[16] P. K. Nagar, D. Savargaonkar, and A. R. Anvikar, "Detection of Dengue Virus-Specific IgM and IgG Antibodies through Peptide Sequences of Envelope and NS1 Proteins for Serological Identification," J Immunol Res, vol. 2020, p. 1820325, Jan. 2020, doi: 10.1155/2020/1820325.

[17] P. Georgiades, Y. Proestos, J. Lelieveld, and K. Erguler, "Machine Learning Modeling of Aedes albopictus Habitat Suitability in the 21st Century," Insects, vol. 14, no. 5, p. 447, May 2023, doi: 10.3390/insects14050447.

[18] K. Mahmood, M. Rashid, S. K. Ansari, F. Kanani, and T. Iftner, "Clinical characteristics of dengue virus infections in Karachi from 2019 to 2023: a cross-sectional study," Sci Rep, vol. 14, no. 1, Dec. 2024, doi: 10.1038/S41598-024-83425-0.

[19] H. Liu et al., "Climate change and Aedes albopictus risks in China: current impact and future projection," Infect Dis Poverty, vol. 12, no. 1, p. 26, Dec. 2023, doi: 10.1186/s40249-023-01083-2.

[20] T. Lura, T. Su, J. Thieme, and M. Brown, "A validated triplex RT-qPCR protocol to simultaneously detect chikungunya, dengue and Zika viruses in mosquitoes," J Vector Borne Dis, vol. 59, no. 3, pp. 198–205, Jul. 2022, doi: 10.4103/0972-9062.316275.

[21] M. Micocci et al., "An easier life to come for mosquito researchers: field-testing across Italy supports VECTRACK system for automatic counting, identification and absolute density estimation of Aedes albopictus and Culex pipiens adults," Parasites and Vectors , vol. 17, no. 1, p. 409, Dec. 2024, doi: 10.1186/s13071-024-06479-z.

[22] F. Sarder, S. Akter, and S. Akter, "Predicting Dengue Outbreak from Climate Data Using Machine Learning Algorithms," IEEE International Conference on Data Science and Information System, ICDSIS 2022, p. Hassan, doi: 10.1109/ICDSIS55133.2022.9915862.

[23] G. Koplewitz, F. Lu, L. Clemente, C. Buckee, and M. Santillana, "Predicting dengue incidence leveraging internet-based data sources. A case study in 20 cities in Brazil," PLoS Negl Trop Dis, vol. 16, no. 1, p. e0010071, Jan. 2022, doi: 10.1371/JOURNAL.PNTD.0010071.

[24] R. Gangula, L. Thirupathi, R. Parupati, K. Sreeveda, and S. Gattoju, "Ensemble machine learning based prediction of dengue disease with performance and accuracy elevation patterns," Mater Today Proc, vol. 80, pp. 3458–3463, Jan. 2023, doi: 10.1016/j.matpr.2021.07.270.

[25] J. K. Chaw et al., "A predictive analytics model using machine learning algorithms to estimate the risk of shock development among dengue patients," Healthcare Analytics, vol. 5, p. 100290, Jun. 2024, doi: 10.1016/J.HEALTH.2023.100290.

[26] N. O. Iheonu, U. K. Nwajeri, and A. Omame, "A non-integer order model for Zika and Dengue co-dynamics with cross-enhancement," Healthcare Analytics, vol. 4, p. 100276, Dec. 2023, doi: 10.1016/J.HEALTH.2023.100276.

[27] M. J. Page et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," BMJ, vol. 372, Mar. 2021, doi: 10.1136/BMJ.N71.

[28] 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: https://doi.org/10.1002/cl2.1230.

[29] X. Guo et al., "Modelling the dynamic basic reproduction number of dengue based on MOI of Aedes albopictus derived from a multi-site field investigation in Guangzhou, a subtropical region," Parasit Vectors, vol. 17, no. 1, p. 79, Feb. 2024, doi: 10.1186/s13071-024-06121-y.

[30] L. Z. Garamszegi, Z. Soltész, K. Kurucz, and T. Szentiványi, "Using community science data to assess the association between urbanization and the presence of invasive Aedes species in Hungary," Parasit Vectors, vol. 16, no. 1, p. 158, May 2023, doi: 10.1186/s13071-023-05780-7.

[31] D. Dharmamuthuraja, R. P. D., I. Lakshmi M., K. Isvaran, S. K. Ghosh, and F. Ishtiaq, "Determinants of Aedes mosquito larval ecology in a heterogeneous urban environment- a longitudinal study in Bengaluru, India," PLoS Negl Trop Dis, vol. 17, no. 11, p. e0011702, Nov. 2023, doi: 10.1371/journal.pntd.0011702.

[32] S.-M. Seyed-Khorami et al., "A comprehensive seroepidemiology of dengue and chikungunya arboviruses in Iran, 2020–2023," Virol J, vol. 21, no. 1, p. 305, Nov. 2024, doi: 10.1186/s12985-024-02574-w.

[33] M. N. P. P. et al., "Island biogeography and human practices drive ecological connectivity in mosquito species richness in the Lakshadweep Archipelago," Sci Rep, vol. 12, no. 1, p. 8060, May 2022, doi: 10.1038/s41598-022-11898-y.

[34] C. D. Griffin, C. Tominiko, M. C. I. Medeiros, and J. W. Walguarnery, "Microplastic pollution differentially affects development of disease-vectoring Aedes and Culex mosquitoes," Ecotoxicol Environ Saf, vol. 267, p. 115639, Nov. 2023, doi: 10.1016/j.ecoenv.2023.115639.

[35] S. S. Sahu, M. M. Baig, D. K. Panigrahi, A. N. Shriram, and A. Kumar, "Atypical Dengue Outbreak in Odisha: Insights from the Entomological Investigations," Indian J Public Health, vol. 68, no. 2, pp. 329–332, Apr. 2024, doi: 10.4103/ijph.ijph_749_23.

[36] N. E. Akyea-Bobi et al., "Entomological risk assessment for transmission of arboviral diseases by Aedes mosquitoes in a domestic and forest site in Accra, Ghana," PLoS One, vol. 18, no. 12, p. e0295390, Dec. 2023, doi: 10.1371/journal.pone.0295390.

[37] P. Duval, C. Valiente Moro, and C. Aschan-Leygonie, "How do attitudes shape protective practices against the Asian tiger mosquito in community gardens in a nonendemic country?," Parasit Vectors, vol. 15, no. 1, p. 439, Nov. 2022, doi: 10.1186/s13071-022-05520-3.

[38] G. Macaluso et al., "First neuroinvasive human case of West Nile Disease in Southern Italy: Results of the 'One Health' approach," Vet Med Sci, vol. 7, no. 6, pp. 2463–2472, Nov. 2021, doi: 10.1002/vms3.591.

[39] D. S. Dissanayake, C. D. Wijekoon, and H. Wegiriya, "Isolation and molecular detection of dengue and chikungunya virus from field-collected adult mosquitoes in Kelantan, Malaysia," J Vector Borne Dis, vol. 61, no. 4, pp. 564–573, Oct. 2024, doi: 10.4103/0972-9062.392269.

[40] J. Tsecouras, W. Walton, R. Schimerlik, and L. W. Cohnstaedt, "Continuing the Role of the Citizen Scientist: Larval & Pupal Collections for National Mosquito Distribution Surveys," Am Biol Teach, vol. 84, no. 4, pp. 195–201, Apr. 2022, doi: 10.1525/abt.2022.84.4.195.

[41] J. Martínez-de la Puente et al., "Do Invasive Mosquito and Bird Species Alter Avian Malaria Parasite Transmission?," Diversity (Basel), vol. 12, no. 3, p. 111, Mar. 2020, doi: 10.3390/d12030111.

[42] P. Farina et al., "Leaf and Flower Extracts from the Dwarf Elder (Sambucus ebulus): Toxicity and Repellence against Cosmopolitan Mosquito-Borne Diseases Vectors," Insects, vol. 15, no. 7, p. 482, Jun. 2024, doi: 10.3390/insects15070482.

[43] M. Sarwar and B. RASOOL, "Seasonal occurrence and biological parameters of Aedes albopictus (Skuse) (Diptera: Culicidae) as management tactics in Faisalabad, Punjab, Pakistan," Rev Soc Entomol Argent, vol. 81, no. 04, pp. 33–41, Dec. 2022, doi: 10.25085/rsea.810405.

[44] S. Sandhya K, S. Sivakumar M, and V. Vijayakumar AR, "Larvicidal Activity and Molecular Docking Studies of Vitexicarpin from Vitex negundo Linn," Journal of Young Pharmacists, vol. 13, no. 3, pp. 223–228, Sep. 2021, doi: 10.5530/jyp.2021.13.46.

[45] K. M. Westby, S. A. Juliano, and K. A. Medley, "Aedes albopictus (Diptera: Culicidae) Has Not Become the Dominant Species in Artificial Container Habitats in a Temperate Forest More Than a Decade After Establishment," J Med Entomol, vol. 58, no. 2, pp. 950–955, Mar. 2021, doi: 10.1093/jme/tjaa215.

[46] B. DEMİRCİ, H. BEDİR, M. ÖZTÜRK, and M. M. AKİNER, "Status of the invasive mosquito species Aedes aegypti (L., 1762) and Aedes albopictus (Skuse, 1895) (Diptera: Culicidae) in Turkey," Turkish Journal of Entomology, vol. 45, no. 2, pp. 269–282, Jun. 2021, doi: 10.16970/entoted.879297.

[47] C. Maichak, K. Hiney, S. R. Loss, J. L. Talley, and B. H. Noden, "Effects of woody plant encroachment by eastern redcedar on mosquito communities in Oklahoma," J Vector Ecol, vol. 47, no. 2, Aug. 2022, doi: 10.52707/1081-1710-47.2.179.

[48] A. DRAGO, G. SPANÒ, G. FACCIONI, and E. MASSELLA, "Olfactory responsiveness of Culex quinquefasciatus and Aedes albopictus (Diptera: Culicidae): Interactions between species, age and attractants," Eur J Entomol, vol. 118, pp. 171–181, Jun. 2021, doi: 10.14411/eje.2021.018.

[49] P. Thongsripong, Z. Qu, J. O. Yukich, J. M. Hyman, and D. M. Wesson, "An Investigation of Human–Mosquito Contact Using Surveys and Its Application in Assessing Dengue Viral Transmission Risk," J Med Entomol, vol. 57, no. 6, pp. 1942–1954, Nov. 2020, doi: 10.1093/jme/tjaa134.

[50] H. Puerta-Guardo et al., "Wolbachia in Native Populations of Aedes albopictus (Diptera: Culicidae) From Yucatan Peninsula, Mexico," Journal of Insect Science, vol. 20, no. 5, Sep. 2020, doi: 10.1093/jisesa/ieaa096.

[51] J. A. Torres-Monzón, M. Casas-Martínez, and T. López-Ordónez, "Infection of Aedes mosquitoes by native Wolbachia in urban cemeteries of Southern Mexico," Salud Publica Mex, vol. 62, no. 4, jul-ago, p. 447, Jun. 2020, doi: 10.21149/10163.

[52] A. N. Tedjou, B. Kamgang, A. P. Yougang, T. A. Wilson-Bahun, F. Njiokou, and C. S. Wondji, "Patterns of Ecological Adaptation of Aedes aegypti and Aedes albopictus and Stegomyia Indices Highlight the Potential Risk of Arbovirus Transmission in Yaoundé, the Capital City of Cameroon," Pathogens, vol. 9, no. 6, p. 491, Jun. 2020, doi: 10.3390/pathogens9060491.

[53] S. Devi et al., "Prevalence of dengue vectors, larval breeding habitats, Stegomyia indices and their correlation with dengue cases in urban and rural areas of Punjab, India," J Vector Borne Dis, vol. 57, no. 2, p. 176, 2020, doi: 10.4103/0972-9062.313966.

[54] M. de B. Arduino, L. F. Mucci, L. M. dos Santos, and M. F. de S. Soares, "Importance of microenvironment to arbovirus vector distribution in an urban area, São Paulo, Brazil," Rev Soc Bras Med Trop, vol. 53, 2020, doi: 10.1590/0037-8682-0504-2019.

[55] G. Hanson et al., "Machine learning and molecular docking prediction of potential inhibitors against dengue virus," Front Chem, vol. 12, Dec. 2024, doi: 10.3389/fchem.2024.1510029.

[56] G. Li, B. Iyer, V. B. S. Prasath, Y. Ni, and N. Salomonis, "DeepImmuno: deep learning-empowered prediction and generation of immunogenic peptides for T-cell immunity," Brief Bioinform, vol. 22, no. 6, Nov. 2021, doi: 10.1093/bib/bbab160.

[57] J. Sotomayor-Bonilla, E. Del Callejo-Canal, C. González-Salazar, G. Suzán, and C. R. Stephens, "Using Data Mining and Network Analysis to Infer Arboviral Dynamics: The Case of Mosquito-Borne Flaviviruses Reported in Mexico," Insects, vol. 12, no. 5, p. 398, Apr. 2021, doi: 10.3390/insects12050398.

[58] F. Deng, S. Wu, Y. Wu, X. Liu, P. Wu, and Z. Zhai, "Identification of mucins and their expression in the vector mosquito Aedes albopictus," Journal of Vector Ecology, vol. 45, no. 2, pp. 297–305, Dec. 2020, doi: 10.1111/jvec.12400.

[59] T. Akter, Md. T. Islam, Md. F. Hossain, and M. S. Ullah, "A Comparative Study between Time Series and Machine Learning Technique to Predict Dengue Fever in Dhaka City," Discrete Dyn Nat Soc, vol. 2024, no. 1, Jan. 2024, doi: 10.1155/2024/2757381.

[60] S. Gautam, A. Thakur, A. Rajput, and M. Kumar, "Anti-Dengue: A Machine Learning-Assisted Prediction of Small Molecule Antivirals against Dengue Virus and Implications in Drug Repurposing," Viruses, vol. 16, no. 1, p. 45, Dec. 2023, doi: 10.3390/v16010045.

[61] G. Gupta et al., "DDPM: A Dengue Disease Prediction and Diagnosis Model Using Sentiment Analysis and Machine Learning Algorithms," Diagnostics, vol. 13, no. 6, p. 1093, Mar. 2023, doi: 10.3390/diagnostics13061093.

[62] S. K. Dey et al., "Prediction of dengue incidents using hospitalized patients, metrological and socio-economic data in Bangladesh: A machine learning approach," PLoS One, vol. 17, no. 7, p. e0270933, Jul. 2022, doi: 10.1371/journal.pone.0270933.

[63] A. Horst et al., "Machine Learning Detects Anti-DENV Signatures in Antibody Repertoire Sequences," Front Artif Intell, vol. 4, Oct. 2021, doi: 10.3389/frai.2021.715462.

[64] Y. Sukama, G. F. Hertono, B. D. Handari, and D. Aldila, "Comparing activation functions in predicting dengue hemorrhagic fever cases in DKI Jakarta using recurrent neural networks," in AIP Conference Proceedings, 2020, p. 020059. doi: 10.1063/5.0030456.

[65] A. Abidemi, R. Ahmad, and N. A. B. Aziz, "Assessing the roles of human movement and vector vertical transmission on dengue fever spread and control in connected patches: from modelling to simulation," The European Physical Journal Plus, vol. 136, no. 11, p. 1192, Nov. 2021, doi: 10.1140/epjp/s13360-021-02195-0.

[66] A. Abidemi, N. A. B. Aziz, and E. Pindza, "Deterministic modelling of optimal control strategies for dengue fever transmission in two interconnected patches," Mathematical Sciences, vol. 18, no. 4, pp. 571–609, Dec. 2024, doi: 10.1007/s40096-023-00517-0.

[67] N. Kaur, S. S. S. A. Rahim, J. J. Jaimin, J. J. F. Dony, K. T. Khoon, and K. Ahmed, "The east coast districts are the possible epicenter of severe dengue in Sabah," J Physiol Anthropol, vol. 39, no. 1, p. 19, Dec. 2020, doi: 10.1186/s40101-020-00230-0.

[68] B. Yuan, H. Lee, and H. Nishiura, "Analysis of international traveler mobility patterns in Tokyo to identify geographic foci of dengue fever risk," Theor Biol Med Model, vol. 18, no. 1, p. 17, Dec. 2021, doi: 10.1186/s12976-021-00149-8.

[69] A. B. Siddique et al., "Assessment of perceived dengue risk and prevention practices among youth in Bangladesh," Sci Rep, vol. 14, no. 1, p. 3940, Feb. 2024, doi: 10.1038/s41598-024-54238-y.

[70] Md. M. Miah et al., "Assessing the impact of climatic factors on dengue fever transmission in Bangladesh," Aerobiologia (Bologna), vol. 40, no. 2, pp. 233–245, Jun. 2024, doi: 10.1007/s10453-024-09814-0.

[71] M. Piaggio, M. Guzman, E. Pacay, J. Robalino, and T. Ricketts, "Forest Cover and Dengue in Costa Rica: Panel Data Analysis of the Effects of Forest Cover Change on Hospital Admissions and Outbreaks," Environ Resour Econ (Dordr), vol. 87, no. 8, pp. 2095–2114, Aug. 2024, doi: 10.1007/s10640-024-00853-2.

[72] M. H. Eldigail et al., "Recent transmission of dengue virus and associated risk Facors among residents of Kassala state, eastern Sudan," BMC Public Health, vol. 20, no. 1, p. 530, Dec. 2020, doi: 10.1186/s12889-020-08656-y.

[73] H. Gomes et al., "Risk of Dengue and tendency map based on geographic localization of cases and vectorial infestation in the North of Brazil," GeoJournal, vol. 88, no. 5, pp. 5259–5269, Jul. 2023, doi: 10.1007/s10708-023-10892-9.

[74] A. Hitakarun, S. Ramphan, N. Wikan, and D. R. Smith, "Analysis of the virus propagation profile of 14 dengue virus isolates in Aedes albopictus C6/36 cells," BMC Res Notes, vol. 13, no. 1, p. 481, Dec. 2020, doi: 10.1186/s13104-020-05325-6.

[75] Y. Chen et al., "Effects of Overwintering on the Survival and Vector Competence of Aedes albopictus in the Urban Life Cycle of Dengue Virus in Guangzhou, China," Virol Sin, vol. 36, no. 4, pp. 755–761, Aug. 2021, doi: 10.1007/s12250-021-00356-0.

[76] L. Kong et al., "Mosquito densovirus significantly reduces the vector susceptibility to dengue virus serotype 2 in Aedes albopictus mosquitoes (Diptera: Culicidae)," Infect Dis Poverty, vol. 12, no. 1, p. 48, May 2023, doi: 10.1186/s40249-023-01099-8.

[77] D. Kobayashi et al., "Comparative analysis of the susceptibility of Aedes aegypti and Japanese Aedes albopictus to all dengue virus serotypes," Trop Med Health, vol. 51, no. 1, p. 61, Nov. 2023, doi: 10.1186/s41182-023-00553-5.

[78] W. Z. Lim, S. W. Chang, and T. C. Teoh, "Pathway analysis of host responses to dengue virus serotype 2 infection and inhibition of viral envelope protein by naringenin from Ganoderma lucidum," J Biosci, vol. 48, no. 4, p. 49, Nov. 2023, doi: 10.1007/s12038-023-00370-2.

[79] S. Gao et al., "Identification and functional analysis of C-type lectin from mosquito Aedes albopictus in response to dengue virus infection," Parasit Vectors, vol. 17, no. 1, p. 375, Sep. 2024, doi: 10.1186/s13071-024-06453-9.

[80] U. Mustafa, K. S. Kreppel, and E. Sauli, "Dengue virus transmission during non-outbreak period in Dar Es Salaam, Tanzania: a cross-sectional survey," BMC Infect Dis, vol. 24, no. 1, p. 1219, Oct. 2024, doi: 10.1186/s12879-024-10109-5.

[81] S. G. Kakarla et al., "Weather integrated multiple machine learning models for prediction of dengue prevalence in India," Int J Biometeorol, vol. 67, no. 2, pp. 285–297, Feb. 2023, doi: 10.1007/s00484-022-02405-z.

[82] A. Sebastianelli et al., "A reproducible ensemble machine learning approach to forecast dengue outbreaks," Sci Rep, vol. 14, no. 1, p. 3807, Feb. 2024, doi: 10.1038/s41598-024-52796-9.

[83] F. Yavari Nejad and K. D. Varathan, "Identification of significant climatic risk factors and machine learning models in dengue outbreak prediction," BMC Med Inform Decis Mak, vol. 21, no. 1, p. 141, Dec. 2021, doi: 10.1186/s12911-021-01493-y.

[84] E. N. Natali et al., "The dengue-specific immune response and antibody identification with machine learning," NPJ Vaccines, vol. 9, no. 1, p. 16, Jan. 2024, doi: 10.1038/s41541-023-00788-7

APPENDIX

TABLE IX. PRISMA CHECKLIST 2020

Section/Topic	Item	Checklist item	Location where item is reported
TITLE			
Title	1	Identify the report as a systematic review.	Page 1
ABSTRACT			
Structured abstract	2	Refer to the PRISMA 2020 checklist for abstracts.	Page 1
INTRODUCTION			
Justification	3	Describe the rationale for the review in the context of existing knowledge.	Page 2
Objectives	4	Provide an explicit statement of the objectives or questions addressed by the review.	Page 2
METHODS			
Eligibility Criteria	5	Specify the inclusion and exclusion criteria for the review and how studies were grouped for synthesis.	Page 6
Sources of information	6	Specify all databases, registries, websites, organizations, reference lists, and other sources searched or queried to identify studies. Specify the date of the last search or query of each source.	Page 5
Search strategy	7	Present the complete search strategies for all databases, records, and websites, including the filters and limits used.	Page 5
Study selection process	8	Specify the methods used to decide whether a study met the review's inclusion criteria, including how many reviewers examined each record and each retrieved report, whether they worked independently, and, if applicable, the details of the automation tools used in the process are collected.	Page 6 and 7
Data extraction process	9	Specify the methods used to collect data from the reports, including how many reviewers collected data from each report, whether they worked independently, any process for obtaining or confirming data from study investigators, and, if applicable, details of automation tools used in the process.	Page 7
List of data	10a	List and define all the results for which data was searched. Specify whether all results that were compatible with each outcome domain in each study were searched (e.g., for all measures, time points, analyses) and, if not, the methods used to decide which results to collect.	Page 7
	10b	List and define all other variables for which data were sought (e.g., participant and intervention characteristics, sources of funding). Describe the assumptions made about any missing or unclear information.	Not applicable
Assessment of the record of bias of the individual studies	11	Specify the methods used to assess risk of bias in the included studies, including details of the tools used, how many review authors assessed each study and whether they worked independently and, if applicable, details of the automation tools used in the process.	Page 8
Measures of effect	12	Specify for each outcome the effect measure(s) (e.g., risk ratio, mean difference) used in the synthesis or presentation of the results.	Not applicable
Synthesis method	13a	Describe the processes used to decide which studies were eligible for each synthesis (e.g., tabulating the characteristics of the study intervention and comparing them to the groups planned for each synthesis (item #5)).	Page 7
	13b	Describe the methods required to prepare data for presentation or synthesis, such as handling missing summary statistics or data conversions.	Page 7
	13c	Describe any methods used to tabulate or visually display the results of individual studies and syntheses.	Page 7 and 8
	13d	Describe any methods used to synthesize the results and provide a rationale for the options. If a meta-analysis was performed, describe the models, methods for identifying the presence and degree of statistical heterogeneity, and the software packages used.	Not applicable
	13e	Describe any methods used to explore possible causes of heterogeneity among study results (e.g., subgroup analysis, meta-regression).	Not applicable
	13f	Describe any sensitivity analyses performed to assess the robustness of the synthesized results.	Not applicable

Assessing the Effectiveness of MCR-KSM for Waiting Waste Reduction: An Empirical Study

Nargis Fatima[1], Sumaira Nazir[2], Suriayati Chuprat[3]
Department of Software Engineering-Faculty of Engineering and Computing, National University of Modern Languages (NUML), Islamabad, Pakistan[1, 2]
Faculty of Artificial Intelligence, University Technology Malaysia (UTM), Kuala Lumpur, Malaysia[3]

***Abstract*—Modern Code Review (MCR) is a well-known and widely adopted quality assurance activity to develop quality software. Although it is a core activity for improving code quality, it generates various types of waste, including waiting waste, defect waste, and composite solution waste. Besides all other wastes, the waiting waste is the most critical one, leading to mental distress, delayed code merges, and project delays. Researchers have made efforts to reduce the production of waiting waste by providing various automated code review tools, techniques and models, one of them is the MCR Knowledge Sharing Model (MCR-KSM). The model claims that it supports sustainable software engineering by minimizing waiting waste reduction during MCR activities. This study aims to evaluate the effectiveness of MCR-KSM with respect to the reduction of waiting waste produced during MCR activities. The experiment methodology is employed for this purpose. This paper presents the experimental investigation approach along with the results. The experiment was conducted in dual sessions with 28 graduate students having similar educational and industrial experience. The tools and techniques, such as SPSS paired t-test and value stream mapping, are used for experimental data management and analysis. The study results revealed that the model significantly helps in the reduction of waiting waste production. The conducted study has implications for investigators to extend the research with different parameters and settings.**

Keywords*—Modern code review; wastes; waiting waste; software quality; automated code review; sustainable software engineering

I. Introduction

Recent software development practices depend on the human aspects of software engineering. For instance, code reviewing [1] that needs verification of code by developers not write it [2]. Code Reviews are the core activity of software development [3]. Besides the quality assurance, the code review is a well-adopted platform for knowledge sharing that can be apparent from pull request discussions on GitHub [4], [5], [6], [7], [8]. Modern Code Review (MCR) is a process where code is submitted, reviewed, discussed and modified before a decision is made on whether to merge it into the main repository or discard the submitted code [1], [9]. Fig. 1 shows the activities involved in the MCR process [10].

As the code review heavily involves the humanoid aspect, specifically for human-to-code and human-to-human interaction for reading, understanding, and providing feedback to modify the code [11], [12]. This interaction and collaboration can result the code review waste production and become inefficient [13] [14]. Waste refers to "Activities that absorb resources and increase cost without adding value". It refers to "Everything that does not make it to the release" The various insights regarding wastes are presented by [15]. The numerous types of waste generated during code review are specifically reported by [13]. The code review wastes can be cognitive load, needless composite solutions, waiting, negative emotions, poor review, poor or delayed feedback, etc. [8], [14], [16], [17].

It is conveyed that the waiting waste is critical and one of the biggest wastes [18], [19], [20]. In code reviews, waiting wastes generated when the author, after submitting the code, waits for reviewers' feedback [21]. In a survey conducted by [22], a survey participant reported, "Usually you write up some code and then you send it out for review, and then about a day later you ping them to remind them ... and then about half a day later you go to their office and knock on their door".

It is also reported that during code review, when the author submits the code, he/she must wait for timely feedback. After waiting for a long time, the request is rejected. Regarding waiting, they conveyed that it is a typical form of disregard and is interactional unfairness. One of the authors in their research discussing the waiting waste reported that "Contributions are ignored unless I beg for attention. I might have time to contribute a minor improvement. I never have time to beg for attention. If you don't want my help, I got the message loud and clear" [23].

Researchers have contributed towards facilitating such review engagement and waiting waste reduction with the aim of improving the feedback process, overall collaboration and effective knowledge sharing [6], [9], [24]. The author of this research study has developed the Modern Code Review-Knowledge Sharing Model (MCR-KSM) to reduce the waiting waste generation. The details regarding the development of MCR-KSM are presented in our previous research work [3], [15]. In our previous work, the MCR-KSM was developed. This study is an extension of our previous work. To develop the MCR-KSM, the included research methodologies were Systematic Literature Review (SLR), along with Expert Review and Delphi Survey. The results of previous work are available at [3], [15]. This study aims to conduct the experimental evaluation of the developed MCR-KSM. The objective of this study is "To evaluate the effectiveness of the Modern Code Review Knowledge Sharing Model to reduce software engineering waiting waste (waiting time)".

The rest of the paper is planned as follows: Section II deliberates the research background and literature review. The experiment design details, including objective, environment, hypothesis, variable, subject selection, instrument and validity evaluation, are given in Section III. The details regarding experiment execution are covered in Section IV. Section V provides the details about the data collection. Section VI highlights the results analysis and discussion. Sections VII-X highlight the conclusion, study limitations, future work suggestions and contribution.

II. Background

Software engineering is a methodical process that aims to create high-quality software within a specified budget and schedule [25]. It involves various underneath systematic activities to achieve its aims. The underneath activities included [26] requirement identification and management, modeling, development, dynamic testing, Static testing (inspection and code reviews). These activities produced numerous wastes [27]. The generated wastes may lead to numerous other issues such as psychological distress, delay in projects, overspending and software malfunctions. When Toyota revolutionized the industry with lean manufacturing, in the 1980's the idea of waste was first presented by [28], [29]. Later in 2000, the lean paradigm was shifted from lean manufacturing to lean software development [20]. Moreover, it is also contended that managing waste generation ensures green and sustainable software development [30]. Various researches were conducted with the aim to identify waste and procedures, tools and methodologies and a model to reduce the waste [18].

In the context of software engineering numerous wastes have been recognized, for instance, "rework", "defects", complexity, cognitive load, and waiting, etc. [18], [28]. As discussed previously, there are multiple underneath activities involved for successful software engineering; each activity produces waste [27]. Code review is an important activity that contributes towards code quality improvement and quality software development [4], [31], [12].

In code review, the reviewer evaluates the source code before uploading it to the version control repository. Code review is supported with the aid of numerous AI-based review tools, such as Quodo merge, Greptle, CodeRabbit, Gerrit, Corbit, Deepsource, PullApprove, for instance, Code flow, such as Gerrit, Review board, Phabricator, etc. [4], [22], [32], [33], [34],[35]. Fig. 1 represents the MCR process workflow.

The code review activity becomes challenging when the team members must confront various types of waste. For instance, Waiting waste, cognitive load, code duplication, poor feedback, negative emotions, poor review comments, Biases, lack of knowledge, poor code understanding, code complexity, ineffective code review tools, poor communication, confusion, lack of motivation to share knowledge, code inconsistency, etc. [13], [36], [37], [38], [39], [40], [41], [42], [43]. It is also reported that waiting waste is the critical waste, and it is conveyed that the waiting waste must be the organization's primary priority if it is to reduce any waste [19], [20].

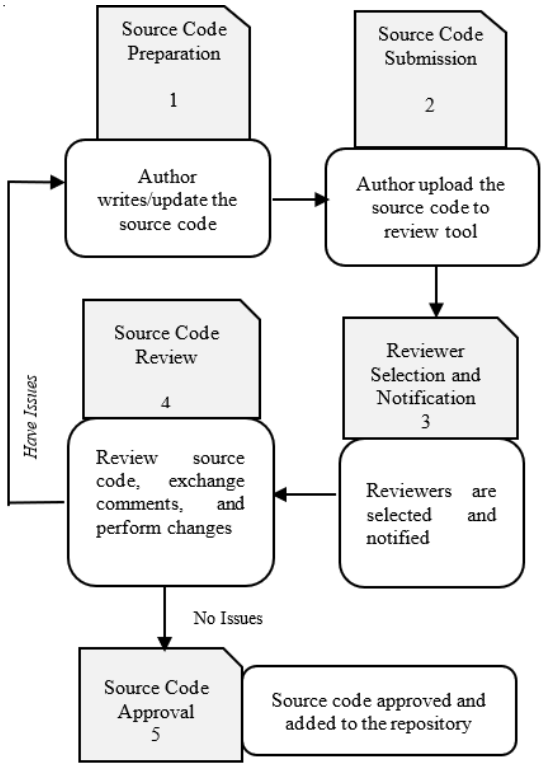


Fig. 1. Workflow of the MCR process [32].

The MCR process allows the freedom for reviewers to accept or reject review requests. This is a critical practice that makes the MCR Process challenging and generates waiting waste. It is reported that as the author submits the code for review, generally the authors wait for the reviewers' response. Numerous research have been performed to highlight the reported issue. A study conducted using dataset of 182 GitHub projects included 55K pull request and 466K code change request comments, and poor clarity of code comments significantly contribute to the delay response [37]. Likewise, poor prioritization of code change requests can contribute to delays and an inefficient review process [14]. A study reported that "Reviewers find only shallow defects and it is hard to give insightful and actionable feedback in timely manner"[44].

Moreover, it is reported that when the author submits the code and request reviewer for review, they have to wait for long time. It became more impertinence when the request is rejected or unnoticed [23]. Likewise, it is conveyed that 16% to 66% of code submitted for review have at least one invited reviewer who let the author in waiting condition [45]. It was argued that regardless of the ease of the MCR process, waiting for feedback is a challenge for authors. It is claimed that multiple human factors such as reviewers' workload, reviewer experience, and familiarity with the requested review can cause poor knowledge sharing and waiting waste [45]. Similarly, it is stated that waiting is the highest reported challenge. In their study a survey participant stated that "Usually you write up some code and then you send it out for review, and then about a day later you ping them to remind them. and then about half a day later you go to their office and knock on their door." [22].

Moreover, It is reported that developers with less experience have to wait for the feedback [32], [46]. Similarly, it is stated that the broadcast or unicast review technology can have a noteworthy effect on delay in response. It is conveyed that broadcast technology has a short response delay as compared to unicast technology [47]. It is stated that there is a need to have guidelines for source code submission to avoid waiting waste [48]. Correspondingly, it is also reported that to get insightful feedback in timely manner is crucial, but the authors often suffer from the non-responsiveness of reviewers and must wait for feedback. The conveyed developer statement is "The value of the feedback is in the proportion to the cost in terms of delay and time spent" [49]. It is argued that developers' reputation, code size are the reason can have impact on timely feedback [50], [51].

Moreover, it is also argued that even in the presence of AI-based code review tools, the wastes are generated during code review [11]. The generated waste can be waiting, rework, defect, needless composite solutions, task switching, code complexity, etc. [28], [43]. It is conveyed that code creation is easy and fast "just prompt LLM to generate desired code", however it is difficult to employ AI-based code review tools to evaluate the code issues. It is reported that expert developers are more valuable than machines and are not replaceable by LLMs [11], [52], [53]. There must be documentation or guidelines to manage the waiting waste and thus project delays [11], [54].

Recent research conveyed that developers' domain knowledge, effective communication and knowledge sharing can reduce waiting waste and project delays [22], [27], [33], [38], [55], [56], [57], [58]. In our previous research work, we have developed the MCR-KSM to reduce waiting waste. In this study, we have presented an experimental study that was performed to validate the designed model with respect to the reduction in waiting waste generation. The upcoming section covers the detailed experiment design, data collection and results analysis.

III. Research Methodology

An experiment in a control environment has been conducted to validate the Modern Code Review Knowledge Sharing Model (MCR-KSM). The aim is to investigate whether the MCR-KSM helps to reduce waiting waste production. The guidelines provided by [59], [60] were utilized for the conduction of the experiment. Grounded on the research work of [61], the experiment was performed with the students of software engineering graduate level having same software development experience and educational background. To conduct the experiment the preparation regarding experiment objectives, selection of environment, hypothesis design, variables, and validity assessment must be completed. This section delivers the particulars regarding planning for the experiment. The experiment execution details are provided in result analysis section. The activities included in the experiment planning are given in sub-sections.

A. Objective of Experiment Conduction

The objective to conduct the experiment was to confirm whether the MCR-KSM help to reduce the waiting waste generation in real environment. The waiting waste production evaluated firstly without using the MCR-KSM, then the waiting waste production was assessed while using the MCR-KSM, assisted by electronic reference guideline of MCR-KSM.

B. Experiment Environment

The experiment environment discusses the context in which the experiment is executed [59]. The experiment was executed in the laboratory of computer science at Comwave Institute of Science and Information Technology, Islamabad, Pakistan. The systems having Windows operating system and C++ editor were utilized to conduct the experiment. The MCR-KSM was also accessible during the experiment.

C. Hypothesis Construction

The experimental statistical analysis is founded on hypothesis testing. A hypothesis is evaluated based on the analysis of dependent variables of the experiment. The guidelines given by [59] were utilized to construct the hypothesis. For experiment pre-test and post-test, the null and alternative hypothesis was constructed. The null hypothesis, H0 refers that there are no real underlying patterns in the experiment situation, the only reasons for differences in observations are coincidental. However, the alternate hypothesis, H1 is in support of which the null hypothesis is rejected.

1) Hypothesis for pre-test: For the pre-test Null and Alternate hypothesis were constructed and are presented in Table I.

TABLE I HYPOTHESIS FOR PRE-TEST

Hypothesis Types	Hypothesis Representation	Hypothesis
Null Hypothesis	H0	There is no noteworthy difference in waiting waste production for both the groups without using MCR-KSM.
Alternate Hypothesis	H1	There is a noteworthy difference in waiting waste production for both the groups without using MCR-KSM.

2) Hypothesis for post-test: For the post-test, hypotheses Null (H_0), Alternate (H_1, and H_2) were constructed. If the H_0 was rejected, it refers to dual situations. The first one was that the waiting waste production in the modern code review process using the MCR-KSM was less than the waiting waste production without using the MCR-KSM. The second situation was that the waiting waste production in the modern code review process using MCR-KSM was greater than the waiting waste production without using MCR-KSM. Therefore, two alternative hypotheses were constructed supporting each situation. Post-test hypothesis are given in Table II.

TABLE II HYPOTHESIS FOR POST-TEST

Hypothesis Types	Hypothesis Representation	Hypothesis
Null Hypothesis	H0	There is no significant difference in the waiting waste production in the modern code review process with or without using MCR-KSM.
Alternate Hypothesis	H1	The waiting waste production in modern code review process using MCR-KSM is lesser than without using MCR-KSM
	H2	The waiting waste generation in modern code review process using MCR-KSM is greater than without using MCR-KSM.

D. Experiment Variables

According to the guidelines provided by [59], independent and dependent variables can be defined for an experiment. In this study, one independent variable was "modern code review process". The process was manipulated with and without the support of MCR-KSM, and the dependent variables were assessed. The dependent variable was the "Waiting Waste". The waiting waste was measured using Value Stream Mapping (VSM) from three aspects, i.e. author waiting time, reviewer waiting time, and total waiting time during MCR activities.

The VSM is utilized by various researchers to calculate the waiting time in various situations. For instance, [29] utilized VSM to measure waiting waste by calculating the customer waiting time, development team waiting time, and total waiting time in the software customization process. Likewise, [20] utilized VSM to measure waiting waste in traditional and agile processes.

In this study, the waiting waste was analysed based on aspects such as "Author Waiting Time (AWT)", that is, waiting time observed by the author while executing MCR activities. "Reviewer Waiting Time (RWT)", that is, the waiting time observed by the reviewer while executing MCR activities and "Total Waiting Time (TWT)", that is, the waiting time observed by the author and reviewers during MCR.

E. Experiment Subject Selection

In this study, 28 students of software engineering were designated as test subjects according to the guidelines given by [62]. The subjects have equal experience. They were grouped as "Group I" and "Group II". Each group contained 14 subjects. Out of 14 subjects in both groups, 7 subjects were designated as "Author" and "Reviewer". The subjects were chosen depending on their industry experience and programming subjects studied during graduation.

F. Experiment Instrument

Formerly, in the experiment implementation, the instruments must be developed. It can be guidelines, objects, and measurement instruments for the measurement [59]. In this experimental study, the included instruments were problem statements that were used by the subjects to write the code, Modern Code Review Knowledge Sharing Model (MCR-KSM), along with an electronic reference guideline, and a document explaining the MCR activities. The details about experiment objective, problem and activities were specified to all the subjects. The subjects "Author" had to write the code, whereas the subjects "Reviewers" had to evaluate the code written by the authors of their group. The subjects of "Group II" were prearranged with the MCR-KSM, supported with its electronic reference guideline. The experiment data were collected via forms as per the directions of [59]. The waiting time experienced by subjects' "Author" and "Reviewers" was measured through the data collected during the experiment. The experiment moderator was given the task of managing the record of the start and end time of each MCR activity.

G. Validity Assessment

It strengthens the rationality of the experimental results. It involves recognition of validity threats, "factors that can influence the dependent variables that are not included as independent variables". The internal and external validity threats were considered as per the guidelines given by [59]. The internal validity threats influence the dependent variable deprived of the researcher's information [59]. The following internal validity threats were discussed for the experiment.

1) Selection effect: It is because of the usual dissimilarity in human recital. Each human can have a different understanding of the English language [59]. It was managed by the selection of subjects with a common educational background. Additionally, they were given similar details about the experiment.

2) Learning effect: It is because of the subjects' behaviour during the experiment activities. It was controlled by common training of the subjects about activities, problems, MCR and MCR-KSM.

3) Instrumentation effect: It is because of the artifacts, such as data collection forms, documents that need to be reviewed, or problems [59]. This hazard was overcome since every subject was given the same task to code, as [59] explained.

4) Information exchange: It is reported that the exchange of information while performing an experiment influences the outcome [59]. It was controlled by strictly observing the subjects during the experiment.

External validity is the external conditions which limit the ability to generalize the experimental results to industrial practice [59]. The two external validity threats, generalizability of subjects and Experiment Scale, were managed. The details about external validity threats are given in sub-sections.

5) Generalizability of subject: It occurs when the population is not taken from the industry [59]. In this study, the selected subjects have industry experience. Thus, this threat was managed as per [59] guideline.

6) Experimental scale: It occurs when the experimental situation or the materials are not illustrative of industrial practice. It was overcome by delivering the subject with real industry problems.

IV. Experiment Execution

The experiment was conducted in compliance with the experiment planning discussed in the previous section, and the actual data was collected and analysed. The experiment was performed in dual sessions. In "Session 1", "Group I" and "Group II" performed MCR activities given in Fig. 1 without the Modern Code Review Knowledge Sharing Model (MCR-KSM). In "Session 2", "Group II" was provided with MCR-KSM with its electronic reference guideline, while "Group I" was not given the model.

Prior to the experiment execution, 28 subjects were distributed into "Group I" and "Group II" as discussed in the design section. The subjects were also provided unique IDs. The "Group I" subjects were provided IDs as "A1 to A7" and "R1 to R7" for the author and the reviewer roles, respectively. Similarly, the "Group II" subjects were given IDs as "A8 to A14" and "R8 to R14". The subjects of both groups with the author role were given a problem statement in both experiment sessions for code writing, whereas the subjects' reviewer role, both groups were given the job to review the code written by the authors of their respective group. The subjects were given 1 hour of training regarding experiment objectives, experiment process, data collection forms, activities, and MCR process. The subjects of "Group II" were described in the electronic guide of the MCR-KSM. In "Session I" of the experiment, no group was given access to MCR-KSM. The "Authors" from "Group I" and "Group II" were requested to write the code in parallel. After finishing the coding activity, they were directed to submit the code to the moderator. After gathering the code from the first subject, "Author", the moderator gave the code to the subject "Reviewer" with the first ID of the same group to review the code. Like this, the "Author-Reviewer" sub-group was established. The "Reviewers" were then asked to review the allocated code and give feedback to the respective "Author". The subject "Author", after receiving the feedback from the subject "Reviewer", was requested to make corrections as suggested by the reviewer. At that moment, the "Author" and "Reviewer" can exchange comments for clarification purposes. The "Author" resubmitted the code to the respective reviewer after addressing the issues. This cycle continues till the chosen reviewer accepts the code. If the reviewer is satisfied with the code, then the code is signed off and formally accepted by the reviewer. During experiment execution, the moderator recorded the waiting time faced by the "Author" or "Reviewer". The data collection details are given in Section V.

V. Data Collection

Based on the experiment design deliberated in Section III, the data were collected. In Session I of the experiment, "Author" from "Group I" and "Group II" were requested to write the code without using MCR-KSM. Similarly, the "Reviewer" from "Group I" and "Group II" were asked to review the code and provide feedback without using MCR-KSM.

During Session II, the subjects "Author" of "Group I" were asked to write the code without using MCR-KSM, and subjects "Authors" of "Group II" were asked to write the code using the MCR-KSM. Likewise, the subjects "Reviewer" of "Group I" were asked to review the code and provide feedback without using MCR-KSM and the subjects "Reviewer" of "Group II" were asked to review the code and provide feedback using MCR-KSM. For both sessions, the waiting time experienced by authors and the reviewers was recorded, and the total waiting time was computed. The collective waiting times for both sessions are given in Tables III and IV. The waiting time is given in Table III, calculated in the experiment session I for the "Groups I" and "Group II" when they executed MCR activities without using MCR-KSM. The waiting time presented in Table IV was computed in the experiment session II when "Group I" performed MCR activities without using the MCR-KSM and 'Group II' performed MCR activities using the MCR-KSM.

Tables III and IV have dual main columns, "Group I" & "Group II". Each main column has four sub-columns. The "Test Subject Sub-group ID" column shows the sub-group ID, the "Group Members" column shows the group members, "Author Waiting Time" column shows the waiting time faced by the author. The "Reviewer Waiting Time" column shows the waiting time faced by the reviewer. "Total Waiting Time" is the collective waiting time faced by the author and reviewer. The result analysis based on the data collected is discussed in Section VI.

TABLE III Session I Evaluation of Waiting Waste Without Using MCR-KSM (Pre-Test)

Group I					Group II				
Test Subjects Sub-group (sg) ID	Group Members	Author Waiting time	Reviewer Waiting time	Total Waiting Time (min)	Test Subjects Sub-group (sg) ID	Group Members	Author Waiting time	Reviewer Waiting time	Total Waiting Time (min)
TSG-I-sg1	A3, R1	40	25.00	60	TSG-II-sg1	A8, R9	42	20	62
TSG-I-sg2	A4, R2	45	35.00	80	TSG-II-sg2	A10, R12	45	45	89
TSG-I-sg3	A2, R3	40	35.00	75	TSG-II-sg3	A9, R10	45	35	90
TSG-I-sg4	A5, R4	55	23.00	78	TSG-II-sg4	A13, R13	45	20	65
TSG-I-sg5	A1, R5	42	32.00	69	TSG-II-sg5	A12, R14	33	43	66
TSG-I-sg6	A6, R6	40	21.00	60	TSG-II-sg6	A1, R11	37	22	59
TSG-I-sg7	A7, R7	39	40.00	79	TSG-II-sg7	A11, R18	40	35	75

VI. Results and Discussion

This section provides the details about the result analysis performed based on the data collected and presented in Section V. The waiting waste, i.e. waiting time was calculated utilizing value stream mapping technique based on variables, i.e. "Author Waiting Time (AWT)", "Reviewer Waiting Time (RWT)", and "Total Waiting Time (TWT)". The collected data is given in Tables III and IV. The experiment data was recorded in SPSS for data analysis. To measure the waiting waste generation in "Pre-test" paired sampled t-test was used [59]. Waiting waste was calculated with three perspectives, i.e. waiting time faced by subject "Authors", subject "Reviewers", and "Total Waiting Time". The mean differences between the dependent variables for examining the waiting waste generation were analyzed. The result analysis of the dependent variables is provided in sub-sections.

1) Pre-test result analysis: By using "paired t-test" the waiting waste production in Pre-test for "Group I" and "Group II was analysed. The subjects were provided with a pre-test where none of the group were given MCR-KSM. Table V shows the stats of the paired t-test for both the groups. The result analysis of the paired t-test shows that there was no considerable difference amongst Group I and Group II who completed experimental activities in pre-test.

The mean values for "Author Waiting Time (Group I, mean=43.00, standard deviation =5.66 Group II, mean =41.00, standard deviation=4.65), Reviewer Waiting Time (Group I, mean=30.1429, standard deviation=7.17469 Group II, mean =31.4286, standard deviation=10.75263) and Total Waiting Time (Group I, mean=71.5, standard deviation=8.695 Group II, mean =72.2, standard deviation=12.75035)" shows that there is no considerable difference in waiting waste production (Author Waiting Time, Reviewer Waiting Time, and Total Waiting Time) for both the groups. The p-value (sig, (2-tailed)) was ($p>0.05$) for author waiting time ($p=0.386$), reviewer waiting time ($p=0.630$), and total waiting time ($p=0.843$). If $p>0.05$ it infers that the null hypothesis was accepted as there was no substantial difference between author waiting time, reviewer waiting time, and total waiting time of "Group I" and "Group II" in Pre-test. It was analyzed from the paired t-test results with variables author waiting time, reviewer waiting time, and total waiting time that there was no significant difference in waiting waste generation when both the group were not provided with Modern Code Review Knowledge Sharing Model. The author waiting time of both groups was almost the same ("Group I", mean=43, "Group II", mean=41). The measure of 2-tailed t-test found this difference not to be significant, $t(7) = 0.935$, $p>0.386$. Hence, it was determined that

> *"The author waiting time was almost same when both the groups were not provided with modern code review knowledge sharing model".*

TABLE IV Session II Evaluation of Waiting Waste Without and With Using MCR-KSM (Post-Test)

Group I (Without (WOT) using modern code review knowledge sharing model)					**Group II (With (WT) using modern code review knowledge sharing model)**				
Test Subjects Sub-group (sg) ID	Group Members	Author Waiting time	Reviewer Waiting time	Total Waiting Time (min)	Participants Sub-group (sg) ID	Group Members	Author Waiting time	Reviewer Waiting time	Total Waiting Time (min)
TSG-I-sg1	A3, R1	37	19	56	TSG-II-sg1	A11, R8	25	15	40
TSG-I-sg2	A4, R2	45	35	80	TSG-II-sg2	A10, R9	20	13	33
TSG-I-sg3	A2, R3	40	35	75	TSG-II-sg3	A9, R10	25	15	40
TSG-I-sg4	A5, R4	55	25	80	TSG-II-sg4	A8, R11	32	16	48
TSG-I-sg5	A1, R5	40	26	66	TSG-II-sg5	A14, R12	30	14	44
TSG-I-sg6	A6, R6	39	20	59	TSG-II-sg6	A12, R13	20	17	37
TSG-I-sg7	A7, R7	36	35	71	TSG-II-sg7	A13, R14	20	10	30

TABLE V Waiting Waste Analysis Pre-Test – Experiment Session I

Variables	**Mean**	**N**	**t**	**Sig. (2-tailed)**
Author Waiting Time (AWT)	43.00	7	0.935	0.386
	41.00	7		
Reviewer Waiting Time (RWT)	30.14	7	-.508	0.630
	31.4286	7		
Total Waiting Time (TWT)	71.5	7	-.207	0.843
	72.2	7		

Similarly, the reviewer waiting time of both the groups was almost the same ("Group I", mean=30.14, "Group II", mean=31.4). The measure of 2-tailed t-test found this difference not to be noteworthy, t (7) = -0.508, p>0.630. Hence, it was concluded that

"The reviewer waiting time was almost same when both the groups were not provided with modern code review knowledge sharing model"

Likewise, the total waiting time of both the groups was almost the same ("Group I", mean=71.5, "Group II", mean=72.2). The measure of 2-tailed t-test found this difference not to be momentous, t (7) = -0.207, p>0.843. Hence, it was concluded that

"The total waiting time was almost same when both the groups were not provided with modern code review knowledge sharing model".

Based on the independent results of dependent variable "author waiting time", "reviewer waiting time", and total waiting time it was concluded that there was no noteworthy difference in the waiting waste generation when both the groups were not provided with the Modern Code Review Knowledge Sharing Model, therefore the null hypothesis, H0, i.e. "There is no noteworthy difference in waiting waste generation for both the groups without using modern code review" was accepted. The paired sample t-test results rejected the alternate hypothesis, H2, i.e. "There is a noteworthy difference in waiting waste generation for both the groups without using modern code review." It was analysed from the results analysis of Group 1 and Group II pre-test that the variance between "Group I" and Group II" was not noteworthy that signifies both the groups were equal regarding their programming capability.

2) Post-test result analysis: The paired t-test was performed to analyze the waiting waste generation for "Group I" and "Group II" in the post-test. The subjects were provided the post-test in which "Group I" was not offered with MCR-KSM and "Group II" was provided with MCR-KSM. Table VI shows the stats of the paired t-test for "Group I" and "Group II" for variables "Author Waiting Time", "Reviewer Waiting Time", and "Total Waiting time".

The stats of paired t-test exhibited that there was a noteworthy variance between "Group I" and "Group II". The mean values for Author Waiting Time "Group I, mean = 41.7143, standard deviation = 6.52468. Group II, mean = 24.5714, standard deviation = 4.96176". Reviewer Waiting Time "Group I, mean = 27.0571, standard deviation = 7.12808 Group II, mean = 14.2857, standard deviation = 2.28869" and Total Waiting Time "Group I, mean = 69.5714, standard deviation = 9.64118 Group II, mean = 38.8571, standard deviation = 6.17599" exhibited a noteworthy difference in waiting waste generation for both the groups.

TABLE VI WAITING WASTE ANALYSIS POST-TEST – EXPERIMENT SESSION II

Group ID	Variables	Mean	N	t	Sig. (2-tailed)
Group I	Author Waiting Time (AWT)	41.7143	7	8.216	0.000
Group II		24.5714	7		
Group I	Reviewer Waiting Time (RWT)	27.0571	7	4.058	0.007
Group II		14.2857	7		
Group I	Total Waiting Time (TWT)	69.5714	7	7.223	0.000
Group II		38.85	7		

Therefore, it was analysed that there was a noteworthy difference regarding waiting waste generation "Author Waiting Time, Reviewer Waiting Time, and Total Waiting Time" amongst "Group I" and "Group II". The p-value (sig, (2-tailed)) was (p<0.05) for author waiting time (p=0.00), reviewer waiting time (p=0.007), and total waiting time (p=0.000). The p value i.e. p<0.05 it infers that the null hypothesis "There is no difference in the waiting waste generation in the modern code review process with or without using Modern Code Review Knowledge Sharing Model" was rejected and there was a noteworthy difference between author waiting time, reviewer waiting time, and total waiting time of "Group I" and "Group II". The positives t values for Author Waiting Time (t=8.216), Reviewer Waiting Time (t=4.058), and Total Waiting Time (t=7.223) indicate that the Modern Code Review Knowledge Sharing Model (MCR-KSM) reduces the author waiting time, reviewer waiting time, and total waiting time. From the paired t-test results it was obvious that that there was a noteworthy decrease in waiting waste generation when participants considered the MCR-KSM.

The waiting time of the "Group II" author was less when they performed MCR activities while using MCR-KSM (mean = 24.5714, standard deviation = 4.96176) as compared to "Group I" author who did not use MCR-KSM (mean = 41.7143, standard deviation = 6.52468). Measure of 2-tailed t-test found this difference to be noteworthy, t (7) = 8.216, p<0.05. Hence, it was determined that

"The author waiting time was reduced with the use of modern code review knowledge sharing model".

Fig. 2 shows the comparative view of Author Waiting Time with and without using MCR-KSM during Post-test session. It clearly shows that when code review was performed using MCR-KSM the average Authors Waiting Time (AWT) was significantly less, i.e. 24.5 (min) as compared to code review activities when performed without using MCR-KSM where average AWT was 41.7 (min).

Likewise, waiting time of the "Group II" reviewer was less when they performed MCR activities while using MCR-KSM (mean = 27.0571, standard deviation = 7.12808) as compared to "Group I" who did not MCR-KSM (mean = 14.2857, standard deviation = 2.28869). The measure of 2-tailed t-test

found this difference to be significant, t (7) = 4.058, p<0.05. Hence, it was determined that

"The reviewer waiting time was reduced with the use of modern code review knowledge sharing model".

Fig. 3 shows the comparative view of Reviewer Waiting Time with and without using MCR-KSM during Post-test session. It evidently shows that when code review was performed using MCR-KSM the average Reviewer Waiting Time (RWT) was significantly less, i.e. 14.2 (min) as compared to code review activities when performed without using MCR-KSM where average RWT was 27.0 (min).

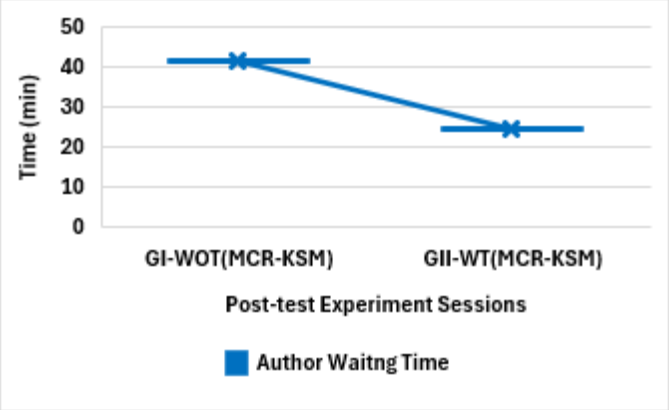


Fig. 2. Comparative view of author waiting time (AWT)-post-test.

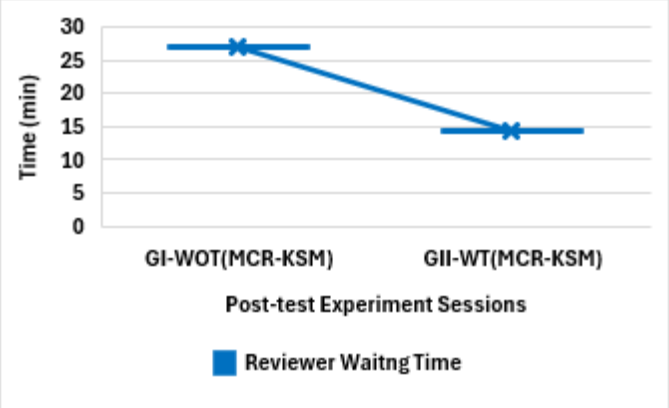


Fig. 3. Comparative view of reviewer waiting time (RWT)-post-test.

Equally, total waiting time was less for "Group II" when the subjects performed MCR activities while using MCR-KSM (mean = 38.8571, standard deviation = 6.17599) as compared "Group I" to when they did not use the MCR-KSM (mean = 69.5714, standard deviation = 9.64118). The measure of 2-tailed t-test found this difference to be noteworthy, t (7) = 7.223, p<0.05. Therefore, it was determined that

"the total waiting time was reduced with the use of modern code review knowledge sharing model".

Fig. 4 shows the comparative view of Total Waiting Time with and without using MCR-KSM during Post-test session. It clearly shows that when code review was performed using MCR-KSM the average Total Waiting Time (TWT) was significantly less i.e.38.8 (min) as compared to code review activities when performed without using MCR-KSM where average TWT was 69.5 (min).

Based on the independent results of dependent variable "Author Waiting Time", "Reviewer Waiting Time", and "Total Waiting Time", it was determined that there was a noteworthy variance in the waiting waste generation with and without the support of the MCR-KSM, so the null hypothesis, H0, i.e. "There is no difference in the waiting waste generation in the modern code review process with or without using Modern Code Review Knowledge Sharing Model" was rejected. The paired sample t-test results supported the alternate hypothesis, H1, i.e.

"The waiting waste generation in MCR using MCR-KSM is lower than without using MCR-KSM.".

The alternate hypothesis H_2 given in Table II was not supported by the paired sample t-test.

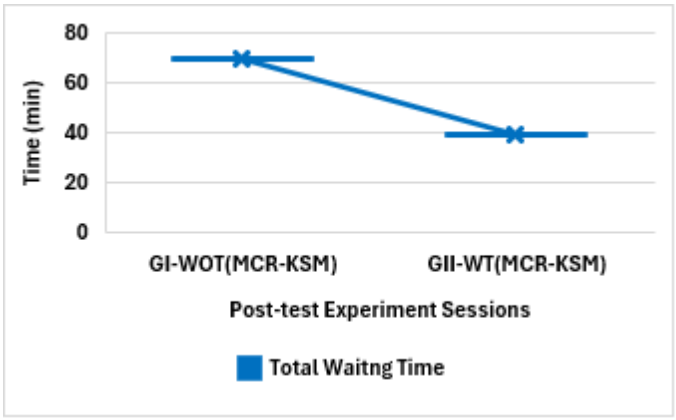


Fig. 4. Comparative view of waiting waste generation-post-test.

VII. Conclusion

MCR is a quality assurance activity, though it is supported with AI and LLM, but it generates various wastes such as rework, defect waste, composite solution waste and waiting waste, etc. Waiting waste is the critical one that causes mental distress. Reducing waiting waste is crucial for sustainable software engineering. The Knowledge exchange has a profound role in the reduction of waiting waste production during code reviews. To reduce waiting waste, our previous study introduced the Modern Code Review Knowledge Sharing Model (MCR-KSM). This study presented empirical results to evaluate the effectiveness of MCR-KSM for the reduction in waste generation during the MCR process. The waiting waste was calculated based on variables, i.e. "Author Waiting Time" and "Reviewer Waiting Time". The study results show that there was a significant reduction in the authors, reviewers and total waiting time when they considered the Modern Code Review Knowledge Sharing Model.

VIII. Limitations

To conduct the study, it was hard to gather dedicated respondents. Twenty-eight graduate students with software engineering experience and knowledge of MCR, software engineering wastes, took part in the experiment. The results might be more confident if more students contributed to the experiment. Nevertheless, despite the limitations, we believe that the results have educational and practical implications.

IX. Future Work Opportunities

The experiment can be augmented in future with varying settings. In future, it is planned to evaluate the significance of MCR-KSM in the actual industry environment. Likewise, more experiments can be conducted to check whether the developed model can reduce other waste, such as defect waste and motion waste, negative emotion waste, etc. and emotional intelligence. Moreover, innovation in AI, machine learning, and LLM-based code review is evolving, and it is understood that developers' domain knowledge is essential and cannot be ignored. Therefore, solutions combined with AI, machine learning and developers' domain knowledge can be beneficial for the reduction of various types of code review wastes. Moreover, other software engineering activities or computing domains can be explored for waste identification and reduction.

X. Contribution

The research conducted has created the foundation for green and sustainable computing by developing and evaluating the MCR knowledge sharing model to reduce waste. The study contributed to three aspects. Firstly, it confirms that the MCR-KSM supports in reduction of waiting waste production. Secondly, the research article provided a detailed experiment design and conduct procedure that can be beneficial for young researchers to experiment with their respective research field. Thirdly, the research highlighted future research venues that can be beneficial for practitioners and researchers.

References

[1] S. Nazir, N. Fatima, and S. Chuprat, "Modern code review benefits-primary findings of a systematic literature review," in ACM International Conference Proceeding Series, 2020, pp. 210–215.

[2] T. Maikantis, I. Natsiou, A. Ampatzoglou, A. Chatzigeorgiou, S. Xinogalos, and N. Mittas, "What you See is What you Get: Exploring the Relation between Code Aesthetics and Code Quality," in Proceedings - 2024 ACM/IEEE International Conference on Technical Debt, TechDebt 2024, 2024, pp. 1–10.

[3] N. Fatima, S. Nazir, and S. Chuprat, "Knowledge sharing factors for modern code review to minimize software engineering waste," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 1, pp. 490–497, 2020.

[4] A. Bacchelli and C. Bird, "Expectations, outcomes, and challenges of modern code review," in Proc. International Conference on Software Engineering, 2013, pp. 712–721.

[5] M. Caulo, B. Lin, G. Bavota, G. Scanniello, and M. Lanza, "Knowledge transfer in modern code review," in IEEE International Conference on Program Comprehension, 2020, pp. 230–240.

[6] A. Bouraffa, Y. D. Pham, and W. Maalej, "How do Developers Use Code Suggestions in Pull Request Reviews?," 2025, pp. 227–238.

[7] M. T. Rahman, R. Singh, and M. Y. Sultan, "Automating Patch Set Generation from Code Reviews Using Large Language Models," in Proceedings - 2024 IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI, CAIN 2024, 2024, pp. 273–274.

[8] S. Ahmed and N. U. Eisty, "Exploring the Advances in Identifying Useful Code Review Comments," in International Symposium on Empirical Software Engineering and Measurement, 2023.

[9] T. Jetzen, X. Devroey, N. Matton, and B. Vanderose, "Towards Debiasing Code Review Support," 2025, pp. 143–148.

[10] N. Davila, J. Melegati, and I. Wiese, "Tales From the Trenches: Expectations and Challenges From Practice for Code Review in the Generative AI Era," IEEE Softw., vol. 41, no. 6, pp. 38–45, 2024.

[11] F. Kazemi, M. Lamothe, and S. Mcintosh, "Interrogative Comments Posed by Review Comment Generators: An Empirical Study of Gerrit," in International Symposium on Empirical Software Engineering and Measurement (ESEM 2025), 2025.

[12] S. Nazir, N. Fatima, and S. Chuprat, "Situational factors for modern code review to support software engineers' sustainability," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 1, pp. 498–504, 2020.

[13] C. S. Fatima, Nargis, Nazir, Suimaira, "Software Engineering Wastes – A Perspective of Modern Code Review," in The 3rd International Conference on Software Engineering and Information Management (ICSIM), 2020.

[14] L. Yang et al., "Prioritizing code review requests to improve review efficiency: a simulation study," Empir. Softw. Eng., vol. 30, no. 1, 2025.

[15] N. Fatima, S. Nazir, and S. Chuprat, "Knowledge sharing framework for modern code review to diminish software engineering waste," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 6, pp. 442–450, 2020.

[16] M. S. Rahman, Z. Codabux, and C. K. Roy, "Investigating the Understandability of Review Comments on Code Change Requests," 2025 IEEE/ACM 22nd Int. Conf. Min. Softw. Repos., pp. 539–551, 2025.

[17] Z. Yang et al., "A Survey on Modern Code Review: Progresses, Challenges and Opportunities," vol. 1, no. 1, pp. 1–62, 2024.

[18] H. Alahyari, T. Gorschek, and R. Berntsson, "An exploratory study of waste in software development organizations using agile or lean approaches: A multiple case study at 14 organizations," Inf. Softw. Technol., vol. 105, no. 7, pp. 78–94, 2019.

[19] J. Urrego, R. Munoz, M. Mercado, and D. Correal, "Archinotes: A global agile architecture design approach," Lect. Notes Bus. Inf. Process., vol. 179 LNBIP, pp. 302–311, 2014.

[20] M. Poppendieck and T. Poppendieck, Lean software development: An agile toolkit. 2003.

[21] N. Davila, J. Melegati, and I. Wiese, "Tales from the Trenches: Expectations and Challenges from Practice for Code Review in the Generative AI Era," IEEE Softw., vol. PP, pp. 1–8, 2024.

[22] L. MacLeod, M. Greiler, M. A. Storey, C. Bird, and J. Czerwonka, "Code reviewing in the trenches: Challenges and best practices," IEEE Softw., vol. 35, no. 4, pp. 34–42, 2018.

[23] D. M. German, U. Rey, and J. Carlos, "' Was my contribution fairly reviewed ?' A Framework to Study the Perception of Fairness in Modern Code Reviews," in Proc. ACM/IEEE 40th International Conference on Software Engineering Synthesizing, 2018, no. 2, pp. 523–534.

[24] C. Adapa, S. S. Avulamanda, A. R. K. Anjana, and A. Victor, "AI-Powered Code Review Assistant for Streamlining Pull Request Merging," in Proceedings of ICWITE 2024: IEEE International Conference for Women in Innovation, Technology and Entrepreneurship, 2024, pp. 323–327.

[25] P. Bourque and R. E. Fairley, Guide to the software engineering - Body of knowledge. 2014.

[26] S. Nazir, N. Fatima, and S. Chuprat, "Individual Sustainability Barriers and Mitigation Strategies: Systematic Literature Review Protocol," in 2019 IEEE Conference on Open System, ICOS 2019, 2019, pp. 1–5.

[27] T. Sedano and P. Ralph, "Software Development Waste," in Proc. IEEE/ACM 39th International Conference on Software Engineering, 2017.

[28] N. Fatima, S. Nazir, and S. Chuprat, "Software engineering wastes-A perspective of modern code review," ACM Int. Conf. Proceeding Ser., pp. 93–99, 2020.

[29] S. Mujtaba, R. Feldt, and K. Petersen, "Waste and lead time reduction in a software product customization process with value stream maps," Proc. Aust. Softw. Eng. Conf. ASWEC, pp. 139–148, 2010.

[30] S. R. Ahmad Ibrahim, J. Yahaya, and H. Sallehudin, "Green Software Process Factors: A Qualitative Study," Sustain., vol. 14, no. 18, 2022.

[31] N. Fatima, S. Nazir, and S. Chuprat, "Individual, Social and Personnel Factors Influencing Modern Code Review Process," 2019 IEEE Conf. Open Syst. ICOS 2019, pp. 40–45, 2019.

[32] A. Bosu, J. C. Carver, C. Bird, J. Orbeck, and C. Chockley, "Process aspects and social dynamics of contemporary code review: Insights from open source development and industrial practice at Microsoft," IEEE Trans. Softw. Eng., vol. 43, no. 1, pp. 56–75, 2017.

[33] C. Sadowski, E. Söderberg, L. Church, M. Sipko, and A. Bacchelli, "Modern code review: : A case study at google," in Proc. ACM/IEEE 40th International Conference on Software Engineering: Software Engineering in Practice, 2018, pp. 181–190.

[34] N. Fatima, S. Chuprat, and S. Nazir, "Challenges and Benefits of Modern Code Review-Systematic Literature Review Protocol," in Proc. International Conference on Smart Computing and Electronic Enterprise, 2018, pp. 1–5.

[35] P. J. S. Xuan and Y. Lee, "Automated Code Review and Bug Detection," in 19th International Joint Symposium on Artificial Intelligence and Natural Language Processing, iSAI-NLP 2024, 2024.

[36] Z. Zhang and T. Saber, "Machine Learning Approaches to Code Similarity Measurement: A Systematic Review," IEEE Access, vol. 13. Institute of Electrical and Electronics Engineers Inc., pp. 51729–51764, 2025.

[37] M. S. Rahman, Z. Codabux, and C. K. Roy, "Investigating the Understandability of Review Comments on Code Change Requests," 2025, pp. 539–551.

[38] S. Zamir, A. Rehman, H. Mohsin, E. Zamir, A. Abbas, and F. A. M. Al-Yarimi, "Integrating Pull Request Comment Analysis and Developer Profiles for Expertise-Based Recommendations in Global Software Development," IEEE Access, vol. 13, pp. 16637–16648, 2025.

[39] M. S. S. Chowdhury, M. N. U. R. Chowdhury, F. F. Neha, and A. Haque, "AI-Powered Code Reviews: Leveraging Large Language Models," in IEEE International Conference on Signal Processing and Advance Research in Computing, SPARC 2024, 2024.

[40] G. Rong et al., "Code Comment Inconsistency Detection and Rectification Using a Large Language Model," 2025, pp. 1832–1843.

[41] O. Ben Sghaier, M. Weyssow, and H. Sahraoui, "Harnessing Large Language Models for Curated Code Reviews," 2025, pp. 187–198.

[42] C. Adapa, S. S. Avulamanda, A. R. K. Anjana, and A. Victor, "AI-Powered Code Review Assistant for Streamlining Pull Request Merging," Proc. ICWITE 2024 IEEE Int. Conf. Women Innov. Technol. Entrep., no. Icwite, pp. 323–327, 2024.

[43] S. Kansab, M. Sayagh, F. Bordeleau, and A. Tizghadam, "An Empirical Study on the Amount of Changes Required for Merge Request Acceptance," 2025.

[44] L. Dong et al., "Survey on Pains and Best Practices of Code Review," in Proceedings - Asia-Pacific Software Engineering Conference, APSEC, 2021, vol. 2021-December, pp. 482–491.

[45] S. Ruangwan, P. Thongtanunam, A. Ihara, and K. Matsumoto, "The impact of human factors on the participation decision of reviewers in modern code review," Empir. Softw. Eng. Manuscr., pp. 1–43, 2018.

[46] A. Lee and J. C. Carver, "Are One-Time Contributors Different? A Comparison to Core and Periphery Developers in FLOSS Repositories," Int. Symp. Empir. Softw. Eng. Meas., vol. 2017-Novem, pp. 1–10, 2017.

[47] F. Armstrong, F. Khomh, and B. Adams, "Broadcast vs. unicast review technology: Does it matter?," in in Proc. 10th IEEE International Conference on Software Testing, Verification and Validation, 2017, pp. 219–229.

[48] A. Lee, J. C. Carver, and A. Bosu, "Understanding the impressions, motivations, and barriers of one time code contributors to FLOSS projects: A survey," in in Proce. IEEE/ACM 39th International Conference on Software Engineering, 2017, pp. 187–197.

[49] O. Kononenko, O. Baysal, and M. W. Godfrey, "Code review quality: How developers see it," in Proc. International Conference on Software Engineering, 2016, pp. 1028–1038.

[50] A. Bosu and J. C. Carver, "Impact of developer reputation on code review outcomes in OSS projects," Proc. 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. pp. 1–10, 2014.

[51] A. Bosu, "Modeling modern code review practices in open source software development organizations," in Proc. IDoESE '13 Baltimore, 2013.

[52] G. Foster, "Why AI will never replace human code review," Graphite, 2025. .

[53] M. Vijayvergiya et al., "AI-Assisted Assessment of Coding Practices in Modern Code Review," AIware 2024 - Proc. 1st ACM Int. Conf. AI-Powered Software, Co-located with ESEC/FSE 2024, pp. 85–93, 2024.

[54] Y. B. Alebachew, M. Ko, and C. Brown, "Are We on the Same Page? Examining Developer Perception Alignment in Open Source Code Reviews," Ease'25, 2025.

[55] T. Sedano, "Removing Software Development Waste to Improve Productivity," in Rethinking Productivity in Software Engineering, Apress, 2019, pp. 221–240.

[56] I. Vlachos, E. Siachou, and E. Langwallner, "A perspective on knowledge sharing and lean management: an empirical investigation," Knowl. Manag. Res. Pract., vol. 00, no. 00, pp. 1–16, 2019.

[57] M. Caulo, B. Lin, G. Bavota, G. Scanniello, and M. Lanza, "Knowledge transfer in modern code review," IEEE Int. Conf. Progr. Compr., pp. 230–240, 2020.

[58] A. Serebrenik and A. Bacchelli, "Competencies for Code Review," vol. 7, no. April, pp. 1–33, 2023.

[59] C. Wohlin, P. Runeson, M. Ohlsson C., B. Regnell, and A. Wesslen, Experimentation in software engineering. Springer, 2000.

[60] Kitchenham, S. L. Pfleeger, D. C. Jones, P.W.Hoaglin, K. El Emam, and J. B.A.Rosenberg, "Preliminary guidelines for empirical research in software engineering," 2002.

[61] M. Host, B. Regnell, and C. Wohlin, "Using students as subjects — A comparative study of students and professionals in lead-time impact assessment," Empir. Softw. Eng., vol. 5, pp. 201–214, 2000.

[62] J. C. F. de Winter, "Using the student's t-test with extremely small sample sizes," Pract. Assessment, Res. Eval., vol. 18, no. 10, pp. 1–12, 2013.

A Review of Attention-Enhanced GRU Models with STL Decomposition for Food Loss Forecasting

Ru Poh Tan[1]*, Siew Mooi Lim[2]*, Kuan Yew Leong[3], Shee Chia Lee[4], Siaw Hong Liew[5], Jun Kit Chaw[6]
Faculty of Computing and Information Technology, Tunku Abdul Rahman University of Management and Technology, Kuala Lumpur, Malaysia[1, 2, 4]
Research and Development Department, A.I. System Research Co., Ltd., Kyoto, Japan[3]
Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, Kota Samarahan, Sarawak, Malaysia[5]
Institute of Visual Informatics, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia[6]

***Abstract*—Forecasting food loss with high accuracy is crucial for improving global food security, optimising supply chains, and supporting sustainability goals. However, conventional time series models and standard deep learning techniques, including recurrent neural networks (RNNs), often fall short in handling the irregularity, seasonality, and complexity inherent in food loss data. While Gated Recurrent Units (GRUs) offer advantages over traditional RNNs, such as mitigating vanishing gradients, they still face limitations in modelling long-range dependencies and noisy sequences. This paper reviews recent advancements aimed at overcoming these challenges by enhancing GRU-based models with attention mechanisms and seasonal-trend decomposition using Loess (STL). Evidence from related domains shows that attention mechanisms improve the capture of long-term dependencies and interpretability, while STL decomposition strengthens stability and accuracy by isolating seasonal and trend components. Hybrid GRU models that combine both approaches consistently outperform standalone methods, highlighting their promise for robust and interpretable forecasting. Though underexplored in the context of food loss, this paper identifies the research gap and advocates for domain-specific GRU–attention–STL architectures, offering a foundation for future empirical work to enable timely interventions and foster resilient, data-driven food systems.**

***Keywords*—GRU; food loss forecasting; attention mechanism; seasonal decomposition; STL; loess; time series; deep learning**

I. Introduction

This study tackles the pressing challenge of enhancing GRU-based forecasting for food loss prediction, a task that demands accurate modelling of complex, irregular, and temporally dependent data. Food loss, defined as the measurable reduction in food quantity or quality throughout the supply chain, is not only an economic inefficiency but also a social and environmental concern. Globally, approximately one-third of all food produced—about 1.3 billion tonnes—is lost or wasted every year, representing an estimated USD 1 trillion in direct economic losses and contributing nearly 8–10% of total greenhouse gas emissions [1]. Moreover, about 14% of food is lost between harvest and retail, 32% on food service and more, intensifying the strain on global food systems [2]. These staggering figures highlight the urgency of developing robust forecasting models that can enable timely interventions to minimise loss, improve efficiency, and promote sustainability.

Traditional statistical and deep learning methods have been applied to time series forecasting in various domains, yet they often struggle with the irregularity, seasonality, and noise inherent in food loss data. The shortcomings highlight the need for innovative architectures capable of capturing both fine-grained temporal dynamics and broad structural patterns within food loss datasets, as food loss poses serious implications for food security, economic sustainability [3], and environmental health, necessitating precise, interpretable forecasting models to guide data-driven interventions.

Recent advances in attention mechanisms and decomposition techniques offer promising solutions. Attention mechanisms, such as self-attention, temporal attention, and multi-head attention, dynamically prioritise informative time steps and enhance the interpretability of deep learning models by revealing which patterns contribute most to predictions. At the same time, Seasonal and Trend decomposition using Loess (STL) has proven effective in isolating trend and seasonal components from noisy signals, improving data quality and learning efficiency. However, although advances in deep learning [19], especially RNN-GRU and attention mechanisms, have shown strong potential in time series analysis across various domains [4], the integration of both seasonal-trend decomposition using Loess (STL) and attention in GRU architectures (GRU-attention-STL) for food loss forecasting remain underexplored.

The rationale for this new initiative is therefore twofold: first, to synthesise existing advancements in attention-enhanced GRUs and STL-based decomposition to demonstrate their relevance to food loss forecasting; and second, to advocate for the development of hybrid GRU–attention–STL architectures tailored to the unique challenges of food loss data. Such models are expected to not only boost forecasting accuracy and robustness but also provide interpretable insights into the underlying seasonal and temporal patterns, thereby supporting data-driven decisions in agriculture and supply chain management.

This paper is organised to provide a comprehensive review of recent advancements in GRU-based time series forecasting, with a specific focus on applications relevant to food loss prediction. Section II.A introduces the foundational concepts of GRU models and their role in time series modelling, outlining their structure, strengths, and limitations, particularly in

*Corresponding authors

capturing long-term dependencies and handling complex sequential data. Section II.B explores how attention mechanisms, such as Self-Attention, Bahdanau, Multi-head, Cross-Attention, Dual-Attention, Temporal-Attention and more, have been integrated with GRUs to dynamically prioritise relevant time steps and improve forecasting accuracy in various domains. Section II.C reviews the use of seasonal-trend decomposition using Loess (STL) to preprocess time series data, enabling GRUs to better learn from trend and seasonal patterns by reducing noise and structural complexity. Section III presents the methodological review framework, summarising and comparing existing studies in structured tables based on their use of attention mechanisms and seasonal-trend decomposition using Loess (STL) decomposition, while identifying design patterns, model architectures, and performance outcomes. Section IV discusses the observed relationships across these studies, highlighting how attention mechanisms enhance temporal adaptability and how seasonal-trend decomposition using Loess (STL) supports signal clarity, as well as the synergistic benefits of combining both techniques. Section V outlines key research gaps, emphasising the absence of GRU-attention-STL hybrids in food loss forecasting and posing future research questions around fusion strategies, generalisability, and dataset availability. Finally, Section VI concludes by reaffirming the value of these enhancements for developing more accurate, interpretable, and robust GRU-based forecasting models and encourages future empirical exploration within the food supply chain domain.

II. Literature Review

This section provides an overview of the existing literature related to GRU-based time series forecasting, with a focus on techniques that enhance model performance through attention mechanisms and time series decomposition. These methods have been widely studied and applied across various domains. Their combined application remains underexplored in the context of food loss forecasting. By reviewing key advancements and their outcomes, this literature review aims to highlight the strengths, limitations and potential of these techniques as foundational components for future food loss prediction models.

A. Gated Recurrent Unit in Time Series Forecasting

The GRU [5] is a type of recurrent neural network (RNN) designed for processing sequential data such as time series [6]. While RNNs, Long Short-Term Memory (LSTM) networks [7], and GRUs all model sequential data, they differ in complexity and performance. Standard RNNs often suffer from the vanishing gradient problem, limiting their ability to capture long-term dependencies [8]. LSTMs address this limitation with memory cells and gated mechanisms: input, forget, and output gates that help manage information flow [9]. GRUs, introduced as a simpler alternative to LSTMs, use gating mechanisms as well but with a more compact structure, offering efficient retention of relevant temporal information.

GRUs use only two gates: the update gate, which controls how much past information is retained, and the reset gate, which determines how much past information is forgotten. This streamlined structure enables GRUs to selectively preserve relevant data and discard noise, leading to more efficient learning of sequence patterns. The operations within a GRU cell are defined by equations in (1):

$$z_t = \sigma\,(W_z\,[h_{t-1},\,x_t] + b_z)$$
$$r_t = \sigma\,(W_r\,[h_{t-1},\,x_t] + b_r)$$
$$\widehat{h_t} = tanh(W_{\hat{h}}\,[r_t \circ h_{t-1}, x_t] + b_{\hat{h}})$$
$$h_t = z_t \circ \widehat{h_t} + (1 - z_t) \circ h_{t-1} \quad (1)$$

where z_t and r_t are the update and reset gate vectors respectively at time step t. W's are the learnable weight matrices and the b's are the biases. x_t is the input vector and the h's are the state vectors. $[\cdot,\cdot]$ denotes the concatenation of two vectors into a longer vector. $\tilde{h}_t$ is the candidate output activation while h_t is the updated hidden state or can be said as the final hidden state. $\circ$ denotes element-wise multiplication. For time $t = 0$, the hidden state variable is set to $h_0 = 0$. The update-gate u and reset-gate r, control the flow of information from one time step to another, and thus they are able to capture long-term dependency.

GRUs often match the performance of LSTMs while using fewer parameters and requiring less training time. Their balance of simplicity and effectiveness makes them well-suited for time series forecasting, particularly when short- to medium-term dependencies are key. Fig. 1 illustrates the internal cell structures of RNN, LSTM, and GRU [6].

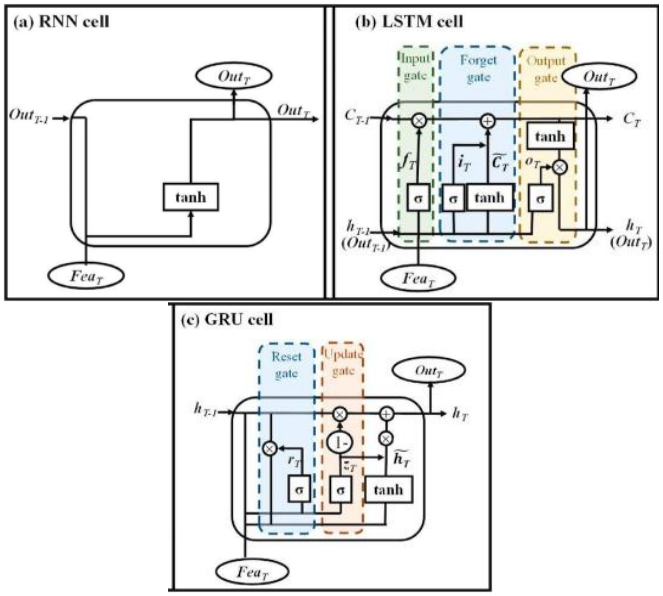


Fig. 1. Internal structure of (a) RNN cell; (b) LSTM cell; (c) GRU cell [6].

GRUs are widely used in time series forecasting for their ability to capture temporal dependencies. They are particularly effective in domains where patterns evolve gradually and depend on past values, such as weather prediction, energy demand, stock markets, and, increasingly, agriculture and food systems.

Fig. 2 shows the unfolded GRU through time [10]. $x_i^{(t)}$ is the input vector at time step vector at time step t. It represents the features or data being fed into the GRU at each time step. $h_i^{(t-1)}$, $h_i^{(t)}$, $h_i^{(t+1)}$, $h_i^{(t+2)}$, and, $h_i^{(t+n)}$ is the hidden state from the previous

time step. Each hidden state captures information from the current input and the previous hidden state. Each box labeled "GRU" processes input x and the previous hidden state h to produce the current hidden state $h_i^{(t)}$.

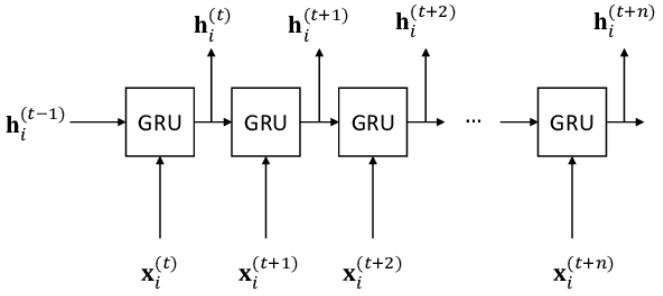


Fig. 2. GRU unfolded through time [10].

GRUs have been applied across various agriculture and food-related forecasting tasks, including crop yield prediction based on environmental data [11], soil moisture estimation [12], food demand forecasting in supply chain [13], spoilage prediction [14], seed trait analysis for protein, oil, and sucrose content [15], and price forecasting of agricultural products [14, 15]. Their ability to handle noisy, nonlinear, and seasonally influenced datasets makes GRUs a valuable tool in agricultural time series modelling.

With fewer parameters and faster training compared to LSTMs, GRUs maintain competitive accuracy [16, 17], particularly for capturing short- to medium-term dependencies, making them suitable for predicting food supply fluctuations and short-term losses. However, GRUs can struggle with highly irregular or seasonal data that require learning long-range dependencies [20]. To address this, recent studies have explored integrating attention mechanisms and time series decomposition to enhance GRU forecasting performance [21].

B. Enhancing GRU with Attention Mechanism

Attention mechanisms significantly enhance sequence learning by allowing models to dynamically focus on the most relevant parts of input sequences. Unlike standard recurrent models that rely on fixed-length context vectors, attention assigns varying weights to different time steps, improving performance on long and complex sequences.

1) Standard attention mechanism: The study [22] introduces a GRU-attention hybrid model for enhancing elastic scaling in container clouds. This model significantly improves prediction accuracy and resource management efficiency in container cloud environments. It effectively handles dynamic loads and reduces operating costs, although further optimisation is needed to control resource oversupply. Fig. 3 shows the prediction curve. Notice that GRU-attention appeared to be more accurate and less error than the GRU-only model.

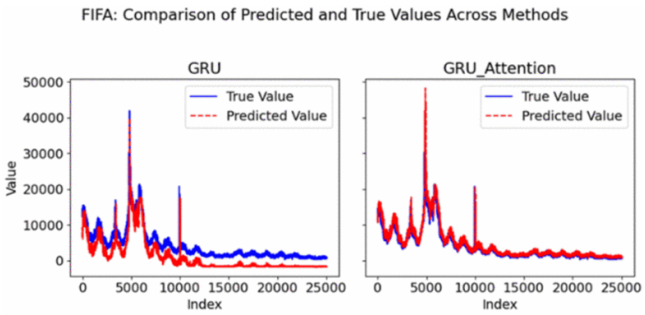


Fig. 3. Comparison of prediction accuracy [22].

To enhance multi-step solar radiation forecasting, Kong et al. [23] used the Kalman Filter [24] for data denoising, and improved input quality. They proposed a hybrid model combining Empirical Mode Decomposition (EMD), GRU, and standard attention mechanisms. Seasonal analysis showed this integration improved accuracy and reduced computational complexity. The structure of the attention-optimised GRU is shown in Fig. 4.

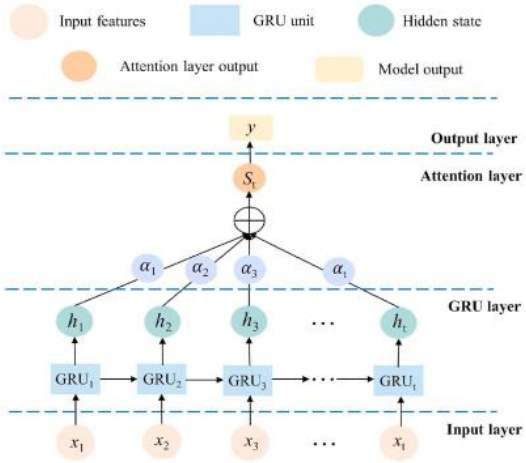


Fig. 4. Architecture diagram of an attention-optimised GRU [23].

2) Cross-attention and multi-scale attention: Cross-attention mechanisms vary in architecture, purpose, and integration. While [23] used standard cross-attention, Farahmand et al. (2025) demonstrate that cross-attention and multi-scale attention in the AttenGluco framework enhance blood glucose forecasting by effectively capturing long-term dependencies, leading to improved prediction accuracy [25]. Zhu et al. [26] also introduced MCI-GRU, a stock prediction model combining multi-head cross-attention with a modified GRU that replaces the reset gate with an attention mechanism to better capture temporal and cross-sectional features.

3) Self-attention and cross-attention: The general formula for the attention is presented in (2):

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

where Q, K, V are the concatenation of query, key and value vectors, and d_k is the dimension of the keys, used for scaling. The softmax makes the scores into probabilities that weight the

values V. In self-attention, the query, key and value all come from the same input sequence (3):

$$Q = XW^{Q}, \quad K = XW^{K}, \quad V = XW^{V} \tag{3}$$

where, X is the input sequence. Self-attention is used to let each position in the input sequence attend to other positions in the same sequence. Cross-attention is similar to self-attention. Self-attention focuses on relationships within a single input sequence, while cross-attention focuses on relationships between two different input sequences (4):

$$Q = X_1W^{Q}, \quad K = X_2W^{K}, \quad V = X_2W^{V} \tag{4}$$

where, X1 is the input where we want to generate attention for example decoder input, while X2 is the input to attend to, the encoder output. In short, the inputs for self-attention are all the same Q = K = V while cross-attention is from different sequences, Q ≠ K and V. Q comes from the decoder's output and K and V come from the encoder's output. The purpose of self-attention is to learn internal dependencies in a sequence. The cross-attention is to learn how one sequence attends to another.

In 2022, Shangzhu Jin et al. [27] enhanced deep knowledge tracing using a GRU network with self-attention, improving sensitivity to long sequences and prediction accuracy by leveraging historical knowledge states. The attention mechanism strengthened the influence of historical knowledge on future performance, leading to higher AUC and precision scores on the ASSISTments datasets compared to benchmark models. This demonstrated significant improvement in knowledge tracing tasks and better prediction performance. In the following year, Xikun Wei et al. [28] applied the Transformer's self-attention to improve runoff prediction in the Yangtze River basin. By integrating self-attention into LSTM and GRU models (LSTM-SA and GRU-SA), they achieved notable accuracy gains, especially under limited training data, by focusing on critical information.

The ProbSparse self-attention mechanism, a variant of traditional self-attention, was introduced to improve efficiency in long-sequence time series forecasting. Study [29] integrates it with LSTM to enhance prediction capacity for such tasks. Given the similarity between LSTM and GRU, integrating ProbSparse self-attention into GRU also holds potential.

Similarly, [30] showed that integrating multi-head self-attention into a GRU framework enhances the modelling of global dependencies and complex temporal patterns. Their FSR-MSAGRU model, applied to PM2.5 forecasting, improved accuracy and generalisation across datasets, reinforcing the value of attention in time-series prediction.

Demertzis et al. [31] showed that integrating GRUs with a cross-modal dynamic attention mechanism improves anomaly detection in time-series data. GRUs enhance prediction accuracy by capturing temporal dependencies and enabling adaptive focus on relevant features, particularly in smart communication environments.

4) Temporal attention and other attentions: Sehovac and Grolinger [32] demonstrated that temporal attention has improved time-series forecasting when combined with GRUs. Their work on electrical load prediction showed that incorporating Bahdanau attention into Seq2Seq RNNs enhances the capture of time dependencies, leading to higher accuracy than traditional models. Similarly, [33] integrated temporal and spatial attention into a GRU-based architecture, significantly boosting forecasting and fault diagnosis for electro-mechanical actuators. Their model, combining seasonal-trend decomposition using Loess (STL), hybrid spatial–temporal attention GRU, and DTW-based similarity, achieved accurate multistep predictions and robust fault detection, outperforming standard GRU and LSTM models. Fig. 5 illustrates the model's structure.

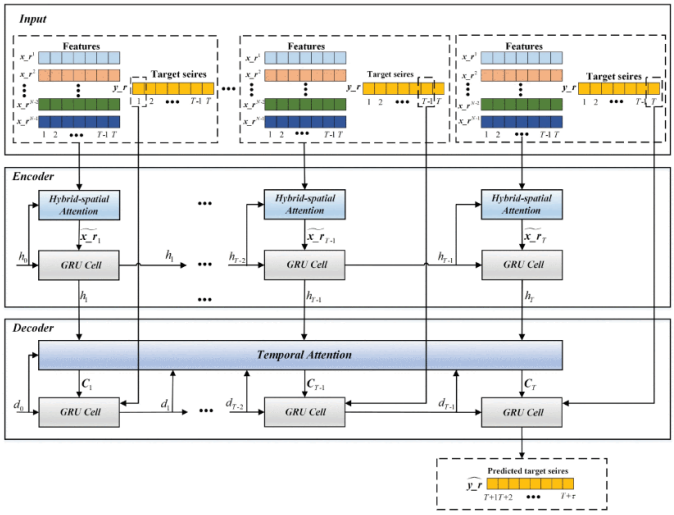


Fig. 5. Structure flowchart of HSTA-GRU model [33].

The authors in [34] investigated integrating GRUs with various attention mechanisms, including Multiplicative, Scaled Dot-Product, Bahdanau, Luong, and Self-Attention, to improve phishing URL detection. After comparing performance across these mechanisms, the study selected top-performing models and benchmarked them against GRU without attention and other baselines. As shown in Fig. 6, attention-enhanced models achieved higher accuracy and overall better performance, indicating that attention helps GRUs focus on relevant input features for more accurate threat detection.

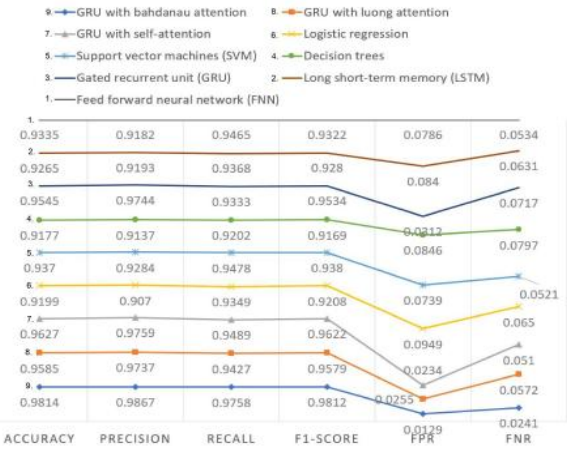


Fig. 6. Comparison of performances with various attentions and models.

The author in [35] introduces a GRU-based encoder-decoder model with a temporal attention mechanism for predicting ship speed over multiple horizons. The temporal attention mechanism and encoder-decoder structure enhance the model's ability to handle multi-horizon predictions and consider exogenous factors. By incorporating exogenous factors and focusing on influential time steps, the model achieves better prediction accuracy. Fig. 7 proves that by adding an attention mechanism to GRU, the average RMSE and average MAE prediction errors are reduced.

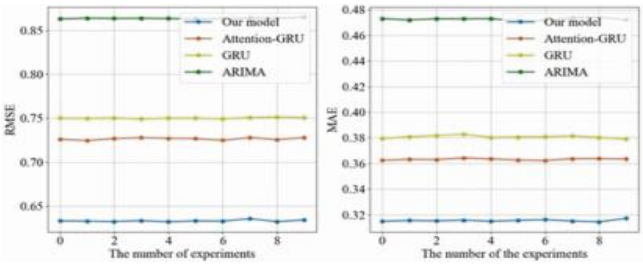


Fig. 7. The RMSE and MAE of different models of the ten experiments [35].

In [36], an interpretable condition monitoring method for offshore wind turbine gearboxes is proposed using a Spatial-Temporal Attention and GRU-based Network (STAGN). The model integrates spatial-temporal attention with GRU to effectively extract features from SCADA data. Fig. 8 shows the structure of the spatial attention module. Each input variable's spatial attention weight is calculated by (5) and (6) is used to calculate the normalised spatial attention weight αj(t) for the j-th input variable at time t.

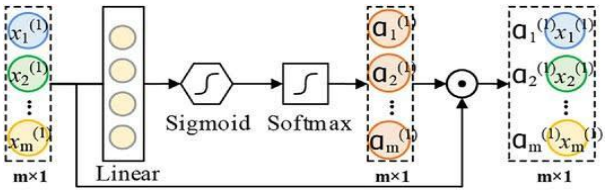


Fig. 8. Structure of spatial attention module [36].

$$e^{(t)} = \sigma(W_{sa}x^{(t)} + b_{sa}) \quad (5)$$

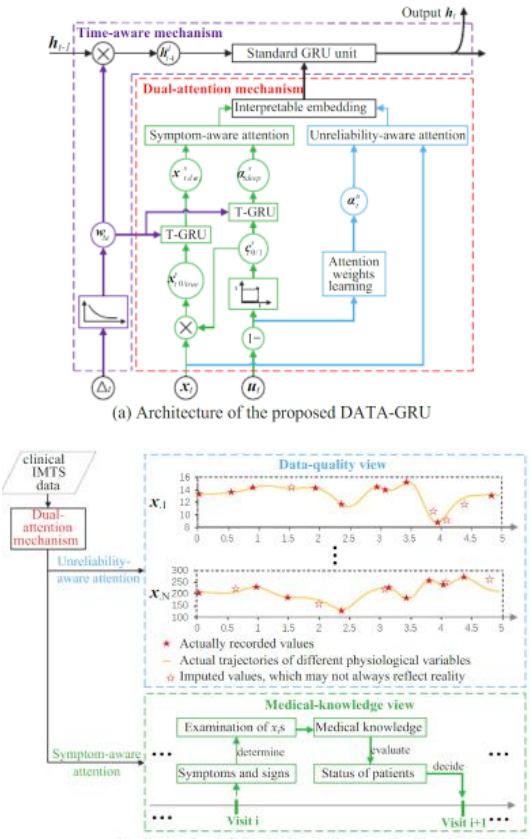
$$\alpha_j{}^{(t)} = \frac{exp(e_j{}^{(t)})}{\sum_{i=1}^{m} e_i{}^{(t)}} \quad (6)$$

According to [36], the temporal attention module enhances key temporal feature representation and mitigates early-stage information loss, while the spatial attention module captures spatial correlations. Together, they improve early fault detection and normal behaviour modelling (NBM), as validated on the Donghai Bridge offshore wind farm. Besides, Cui et al. [37] showed that integrating hierarchical contextual attention into GRU networks improves forecasting in sequential recommendation tasks by effectively capturing both long- and short-term dependencies. Their HCA-GRU model outperformed traditional RNN-based approaches in accuracy and ranking performance on real-world datasets.

5) Trend-aware attention: Trend-aware attention [38] is designed to highlight long-term upward or downward patterns in time-series data, unlike standard attention, which treats all inputs equally. Chen et al. [38] introduced the TAA-Transformer, which incorporates this mechanism to improve lithium-ion battery capacity prediction by capturing both local and global features. Compared to GRU, which lacks trend-aware attention, the TAA-Transformer achieved superior predictive accuracy. Although focused on Transformers, the study suggests that integrating trend-aware attention into GRU is feasible and could enhance performance in similar forecasting tasks.

Trend-aware multi-head flow attention has also been applied to forecasting tasks. Yang et al. [39] proposed JGFACN, a model combining a dynamic Jacobi graph with trend-aware flow attention for traffic prediction. Compared to methods including LSTM [18], JGFACN showed superior performance by dynamically adapting to temporal features. While the study did not integrate attention into GRU, its effectiveness suggests potential gains if applied to GRU-based time-series forecasting models.

6) Dual-attention: Tan et al. [40] also introduced DATA-GRU, a Dual-Attention Time-Aware GRU model, to address irregular intervals and missing values in EHR time-series data. Incorporating unreliability-aware and symptom-aware attention, the model effectively predicts patient mortality risk with improved accuracy and interpretability over state-of-the-art methods. Fig. 9 illustrates the DATA-GRU architecture and attention integration.



(a) Architecture of the proposed DATA-GRU

(b) Illustration of the dual attention mechanism

Fig. 9. Architecture of DATA-GRU is shown in (a) and dual attention mechanism in (b) [40].

The study in [41] proposed a dual-stage attention-based RNN (DA-RNN) using input attention to focus on relevant driving series and temporal attention to capture long-term dependencies, improving time-series forecasting accuracy on datasets like SML 2010 and NASDAQ 100. While based on LSTM, the dual-stage attention mechanism could similarly benefit GRU models. In 2023, [42] introduced PDAGRU, a Parallel GRU model with multiplicative and temporal attention layers, for probabilistic RUL prediction of wind turbine bearings. As shown in Fig. 10, this approach significantly enhances accuracy and reliability by better extracting degradation features from time-series data.

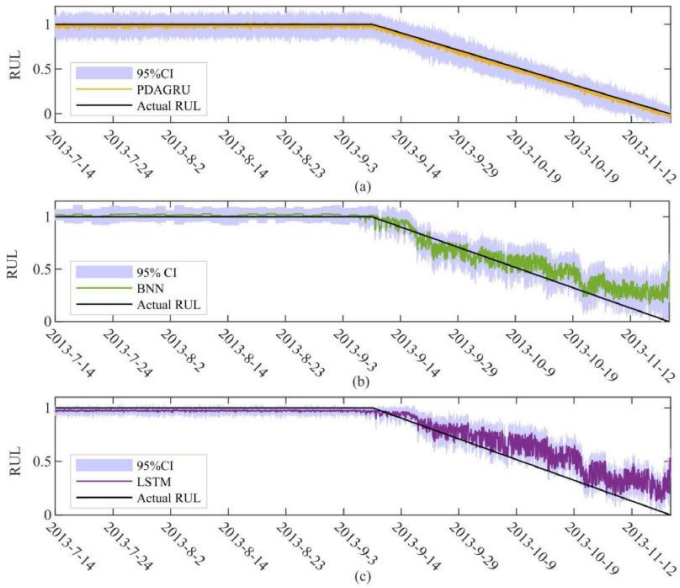


Fig. 10. The RUL prediction performance metrics with different methods [42].

In [43], the Hierarchical Attention Cascade Neural Network (CasHAN) is proposed to predict future cascade growth based on early evolution. As shown in Fig. 11, sequences from cascade graphs are encoded using a bidirectional GRU, then refined through node-level attention (capturing user influence) and sequence-level attention (addressing community redundancy), improving prediction accuracy.

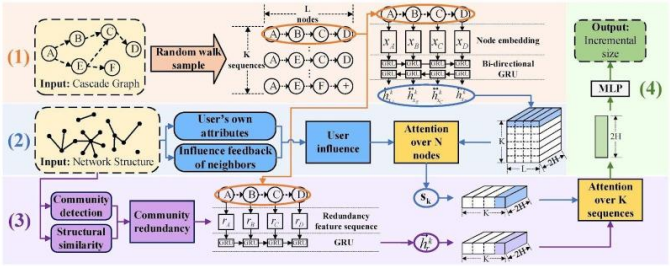


Fig. 11. The overall architecture of the proposed CasHAN model [43].

However, these attention-based GRU models have not been specifically applied to food loss forecasting, which poses distinct challenges due to the irregular, seasonal, and multi-source nature of agricultural data. Unlike applications in stock prediction, solar radiation, or aerospace forecasting, food loss prediction requires models to handle domain-specific variables, temporal variability, and data sparsity. This highlights a critical research gap for attention-enhanced GRU models tailored to food loss forecasting.

C. Enhancing GRU with Seasonal and Trend Decomposition

Seasonal-trend decomposition using Loess (STL) [44] is a time-series technique that separates data into seasonal, trend, and residual components. It uses LOESS, Locally Estimated Scatterplot Smoothing, and a nonparametric regression method, to flexibly estimate trends and seasonality over time. This decomposition aids in forecasting, anomaly detection, and identifying underlying patterns.

Despite advances in time-series modelling, decomposition integration within GRU architectures remains underexplored. Hewamalage et al. [45] noted that while RNNs, including GRUs, can model seasonality, this is mainly effective for time series with consistent seasonal patterns. For nonstationary or variable-seasonality data, they recommend deseasonalisation prior to modelling. Seasonal-trend decomposition using Loess (STL) provides a robust, nonparametric approach to decomposing time series into seasonal, trend, and residual components, enabling models to focus on learning more meaningful patterns from each.

Similarly, [46] proposed a short-term electricity forecasting model combining seasonal-trend decomposition using Loess (STL) with dual GRUs to separately capture global and local dependencies, significantly improving accuracy and robustness in handling nonlinear real-world data. Fig. 12 shows the structure of the proposed scheme.

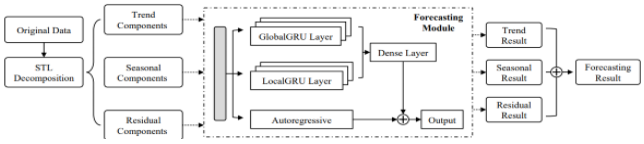


Fig. 12. The structure of the proposed scheme by [46].

The study in [47] introduced a hybrid STL-based framework for rainfall prediction, combining GRU, multi-time-scale GRU, and LightGBM to model decomposed components separately. This approach notably improved forecasting accuracy and robustness, especially for extreme rainfall events, by capturing key patterns in meteorological data. Fig. 13 illustrates the framework of the STL-ML approach to predict rainfall. As noted in [33], seasonal-trend decomposition using Loess (STL) also enhances attention-based GRU models by isolating trend, seasonal, and residual components, enabling more effective pattern extraction and improving fault prediction accuracy in electro-mechanical actuators.

An ensemble model proposed in 2020 [48] applied seasonal-trend decomposition using Loess (STL) with GRUs for base station traffic forecasting, demonstrating that modelling each decomposed component separately and recombining results enhances overall accuracy. Seasonal-trend decomposition using Loess (STL) effectively reduces noise and outliers, allowing GRUs to better capture nonlinear patterns and outperform standalone GRU and SARIMAX models.

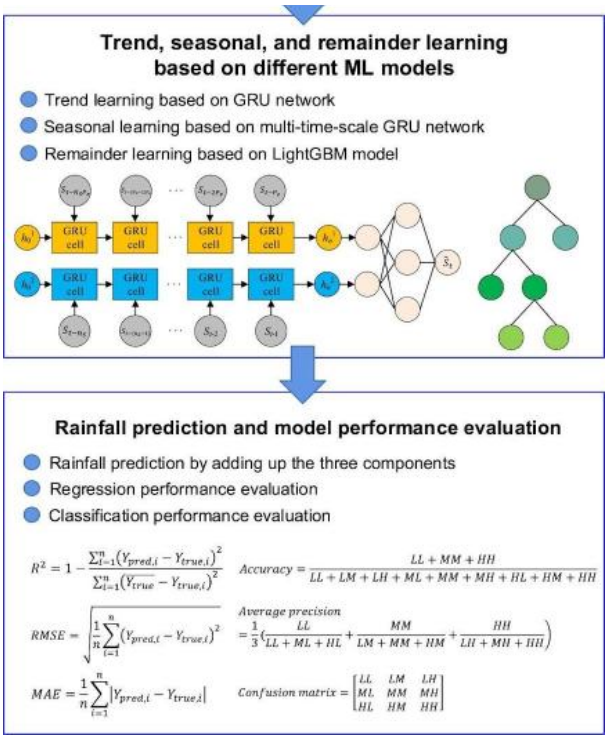


Fig. 13. Framework of the hybrid STL-ML approach for [47].

In climate forecasting, Liu et al. [49] introduced an STL-GRU model for temperature prediction, enabling GRUs to capture complex temporal correlations. This approach improved accuracy and efficiency across multiple regional datasets. The predictive performance indicators comparison for [49] are shown in Fig. 14. Similarly, Jia et al. [50] developed STL-IWOA-GRU for groundwater level prediction, integrating seasonal-trend decomposition using Loess (STL), an improved whale optimisation algorithm (IWOA), and GRU to handle data complexity with strong accuracy and robustness.

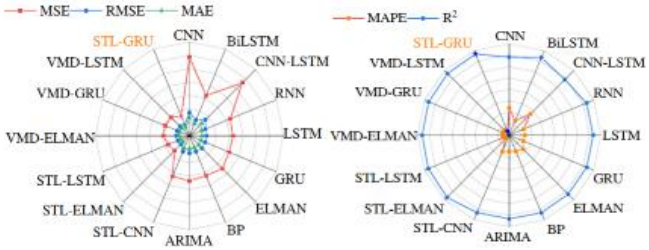


Fig. 14. Predictive performance indicators for [49].

Zhang et al. [51] proposed an STL-GRU-SVR model for sales forecasting, emphasising the importance of modelling trend, seasonal, and residual components independently to improve prediction. Likewise, [30] incorporated seasonal-trend decomposition using Loess (STL) in a GRU-based environmental forecasting model, enhancing feature space reconstruction and improving precision and robustness by isolating periodic and trend components.

III. Methodological Review Framework

This section presents a structured synthesis of existing studies that apply GRU-based models enhanced with attention mechanisms and time series decomposition. To make the findings accessible, the review is organised into a series of tables, each dedicated to a specific category of techniques. For instance, Tables I to VI summarise studies that integrate different types of attention mechanisms into GRU architectures, including Bahdanau (additive) attention, Luong (multiplicative) attention, self-attention, cross-attention, temporal attention, dual-attention, and more advanced designs such as hybrid spatial–temporal and trend-aware attention. Each table outlines the application domain, methodological approach, key contributions, and the specific challenges addressed, allowing us to see at a glance how these mechanisms influence forecasting performance across different contexts. Table VII focuses on studies that incorporate Seasonal-Trend decomposition using Loess (STL), highlighting how decomposing raw data into seasonal, trend, and residual components enhances learning efficiency and accuracy. By categorising and comparing these approaches, the framework provides both technical depth for researchers and clarity for practitioners, illustrating how attention mechanisms and seasonal-trend decomposition using Loess (STL) contribute individually and in combination to improving forecasting outcomes. Below shows review framework for each tables:

- Table I: Standard Attention Mechanisms
- Table II: Cross-Attention and Multi-Scale Attention
- Table III: Self-Attention and Cross-Attention
- Table IV: Temporal Attention and Other Attentions
- Table V: Dual-Attention Reviews
- Table VI: Trend-Aware Attention
- Table VII: Seasonal-Trend Decomposition using Loess (STL)

TABLE I. Review Framework for Standard Attention Mechanisms

Application	Methods	Key contributions	Gap Addressed
Container cloud elastic scaling [22]	GRU-attention hybrid model, Improved Horizontal Pod Autoscaler (HPA)	Enhanced load prediction and elastic scaling in container clouds, improving prediction accuracy and resource management efficiency.	Addressed the challenge of achieving efficient resource scheduling and elastic scaling in dynamic load environments, improving prediction accuracy and adaptability to complex load patterns.
Solar Radiation Prediction	EMD-GRU-Attention (EGA) model combining EMD, GRU, and **standard attention** mechanism.	Enhanced multi-step solar radiation prediction accuracy and reduced computational burden.	Addressed nonlinear, nonstationary data challenges and improved prediction accuracy.

TABLE II. REVIEW FRAMEWORK FOR GRU-CROSS-ATTENTION AND GRU-MULTI-SCALE ATTENTION MECHANISMS

Application	Methods	Key contributions	Gap Addressed
Blood Glucose Forecasting [25]	Multimodal Transformer with **cross-attention** and **multi-scale attention**.	Improved long-term BGL prediction accuracy using AI-READI dataset.	Addressed long-term forecasting challenges and data fusion with different rates.
Stock Prediction [26]	MCI-GRU model with **multi-head cross-attention** and improved GRU.	Enhanced stock prediction accuracy by integrating cross-attention and improved GRU.	Addressed limitations in capturing complex market dynamics and latent states.

TABLE III. REVIEW FRAMEWORK FOR GRU-SELF-ATTENTION AND GRU-CROSS-ATTENTION

Application	Methods	Key contributions	Gap Addressed
Knowledge Tracing [27]	GRU with **Self-Attention**.	Improved prediction accuracy and sensitivity to long sequence data in knowledge tracing.	Addressed information loss in long sequence data and enhanced historical knowledge impact on future performance.
Runoff Prediction in Yangtze River Basin [28]	Transformer (TSF), LSTM, GRU, LSTM-SA, GRU-SA with **Self-Attention**	Demonstrated that GRU outperforms TSF in limited data scenarios; Self-Attention improves LSTM and GRU performance.	Addressed the challenge of limited data for TSF and improved prediction accuracy with Self-Attention in LSTM and GRU.
Long Sequence Time-Series Forecasting [29]	Informer with **ProbSparse self-attention**, self-attention distilling, generative style decoder.	Achieved efficient long sequence forecasting with reduced time complexity and memory usage.	Addressed inefficiencies in Transformer models for long sequence forecasting.
PM2.5 Concentration Prediction [30]	FSR-MSAGRU model using feature space reconstruction and **multihead self-attention** GRU.	Improved prediction accuracy and generalisation by capturing periodic and global features.	Addressed limitations in capturing periodicity and global features in PM2.5 data.
Anomaly Detection in Data Streams [31]	CM-DANA model with **cross-modal dynamic attention** and multimodal learning.	Improved anomaly detection accuracy and efficiency in smart communication environments.	Addressed challenges in handling heterogeneous data types and evolving patterns.

TABLE IV. TEMPORAL ATTENTION AND OTHER ATTENTIONS

Application	Methods	Key contributions	Gap Addressed
Fault Diagnosis for Electro-Mechanical Actuators [33]	STL-HSTA-GRU with Dynamic Time Warping (DTW) for classification. HSTL = hybrid **spatial-temporal attention**	Improved fault diagnosis by combining STL decomposition with hybrid spatial-temporal attention GRU and DTW for classification.	Addressed challenges in handling nonlinear and seasonal data for early fault detection.
Load Forecasting [32]	Sequence-to-Sequence RNN with **Bahdanau Attention**, using GRU, LSTM, and Vanilla RNN cells.	Improved load forecasting accuracy by adapting S2S RNN with attention mechanisms.	Addressed challenges in capturing time dependencies and improving prediction accuracy.
Phishing URL Detection [34]	GRU with various attention mechanisms: **Multiplicative**, **Scaled Dot-Product**, **Bahdanau**, **Luong**, **Self-Attention**.	Achieved 98.14% accuracy using GRU with Bahdanau attention for phishing URL detection.	Enhanced detection accuracy and adaptability over traditional methods, addressing evolving phishing threats.
Ship speed prediction [35]	GRU-based encoder-decoder, **Temporal attention** mechanism, Dropout layers, Huber loss function, Adam optimizer	Proposed a model for accurate and timely ship speed prediction using a GRU-based encoder-decoder with a temporal attention mechanism.	Addressed the challenge of predicting ship speed by considering both previous speed and exogenous factors like weather and sea conditions, and improved prediction accuracy with a temporal attention mechanism.

Offshore Wind Turbine Gearbox Monitoring [36]	**Spatial-Temporal Attention** and Gated Recurrent Unit (GRU)	Introduced an interpretable model using spatial-temporal attention to enhance feature extraction and improve monitoring accuracy.	Addressed the challenge of early-stage information loss and lack of interpretability in condition monitoring models.
Sequential Recommendation [37]	**Hierarchical Contextual Attention**-based GRU (HCA-GRU)	Introduced a hierarchical contextual attention mechanism to enhance short-term interest modelling and combine it with long-term dependencies for better user interest representation.	Addressed the limitation of RNN's monotonic temporal dependency by capturing complex correlations among recent items and assigning nonmonotonic weights.

TABLE V. DUAL-ATTENTION REVIEWS

Application	Methods	Key contributions	Gap Addressed
Mortality Risk Prediction from Irregular Multivariate Time Series [40]	**Dual-Attention Time-Aware** Gated Recurrent Unit (DATA-GRU). The Dual-Attention including **Unreliability-aware attention** and **Symptom-aware attention**.	Introduced a model with time-aware and dual-attention mechanisms to handle irregular intervals and missing values in EHR data.	Addressed the challenges of irregular time intervals and missing values in EHR data by preserving temporal information and improving data reliability.
Time Series Prediction [41]	**Dual-Stage Attention**-Based Recurrent Neural Network (DA-RNN)	Introduced a dual-stage attention mechanism (input and temporal attention) to select relevant input features and capture long-term dependencies in time series.	Addressed the challenge of capturing long-term dependencies and selecting relevant driving series in time series prediction.
Probabilistic RUL Prediction of Wind Turbine Bearings [42]	Parallel GRU with **Dual-Stage Attention** Mechanism (PDAGRU). The Dual-Stage = **multiplicative attention** and **temporal attention**	Introduced a dual-stage attention mechanism to enhance degradation information extraction and a parallel structure for improved prediction accuracy and uncertainty quantification.	Addressed the challenges of efficient degradation information extraction, vanishing gradient problem, and prediction uncertainty.
Information Cascade Prediction [43]	**Hierarchical Attention** Cascade Neural Network (CasHAN) with **Node-Level** and **Sequence-Level Attention**	Introduced a model using hierarchical attention mechanisms to improve cascade prediction by considering user influence and community redundancy.	Addressed the lack of consideration for user influence and community redundancy in cascade prediction models.

TABLE VI. TREND-AWARE ATTENTION

Application	Methods	Key contributions	Gap Addressed
Lithium-Ion Battery Capacity Prediction [38]	**Trend-Aware Attention** (TAA) Transformer with 1-D Convolution.	Introduced TAA to capture local trends and global features, improving prediction accuracy over standard Transformers.	Addressed the limitation of standard Transformers in capturing local trend information in time-series data.
Traffic Forecasting [39]	Dynamic Jacobi Graph and **Trend-Aware Flow Attention** Convolutional Network (JGFACN)	Introduced a model combining dynamic Jacobi graph convolution with trend-aware flow attention to capture spatial-temporal correlations effectively.	Addressed challenges in capturing spatial-temporal correlations, dynamicity, and heterogeneity in traffic data.

Table VII below presents a review of studies that incorporate seasonal-trend decomposition using Loess (STL) as part of their forecasting frameworks. These works demonstrate how it is used to isolate trend and seasonal components from raw time series data, thereby improving model stability and learning efficiency. The table summarises how seasonal-trend decomposition using Loess (STL) has been combined with models such as GRU, LightGBM, and hybrid neural networks across various domains. This synthesis highlights the versatility

and effectiveness of seasonal-trend decomposition using Loess (STL) as a preprocessing technique, offering valuable guidance for its potential application in food loss forecasting.

TABLE VII. REVIEW FRAMEWORK FOR SEASONAL-TREND DECOMPOSITION USING LOESS

Application	Methods	Key contributions	Gap Addressed
Short-Term Electricity Forecasting [46]	Combined GRU with STL Decomposition	Developed a scheme using STL decomposition and dual GRU networks (GlobalGRU and LocalGRU) to capture local and global dependencies, enhancing prediction accuracy.	Addressed the inability of traditional models to capture complex nonlinear relationships and local historical information in electricity data.
Rainfall Time Series Prediction [47]	STL, GRU, Multi-time-scale GRU, LightGBM	Developed a hybrid STL-ML approach integrating STL decomposition with machine learning models to predict rainfall, effectively capturing trend, seasonal, and remainder components.	Addressed the challenge of accurately predicting rainfall by fully extracting and utilising underlying patterns and information from rainfall time series.
Fault Diagnosis for EMAs [33]	STL-HSTA-GRU, SM	Developed a method combining STL for seasonal-trend decomposition with HSTA-GRU for spatio-temporal prediction and SM for classification.	Addressed early-stage fault detection and nonlinear, seasonal data challenges in EMAs.
Base Station Traffic Forecasting [48]	STL, GRU	Developed an ensemble model combining STL decomposition with GRU to improve traffic forecasting accuracy by reducing noise and outliers	Addressed the challenge of nonlinearity and chaotic behaviour in traffic data, enhancing prediction accuracy over standalone models.
		.	
Climate Time Series Forecasting [49]	STL, GRU	Developed a hybrid STL-GRU model to improve temperature prediction accuracy by capturing seasonal variations and long-term trends.	Addressed the challenge of nonlinearity and temporal dependencies in temperature data, enhancing prediction accuracy over traditional models.
Groundwater Level Prediction [50]	STL, IWOA, GRU. IWOA stands for Improved Whale Optimisation Algorithm.	Developed a hybrid STL-IWOA-GRU model to improve GWL prediction accuracy by enhancing convergence speed and global search capabilities.	Addressed the challenge of nonlinearity and complexity in GWL time series under EWR, improving prediction accuracy over traditional models. EWR stands for Ecological Water Replenishment.
Sales Forecasting [51]	STL, SVR, GRU	Developed a sales forecasting model combining STL with SVR and GRU for higher accuracy.	Addressed the lack of direct forecasting using decomposed components in sales data.
PM2.5 Prediction [30]	FSR-MSAGRU (STL, MSAGRU)	Developed a model combining STL for capturing periodic and trend information with feature space reconstruction and multihead self-attention GRU for PM2.5 prediction.	Addressed the neglect of periodic and global features in PM2.5 prediction models.

IV. DISCUSSION

After reviewing a broad range of GRU-based forecasting studies that utilise attention mechanisms and seasonal-trend decomposition using Loess (STL), several clear relationships and patterns emerge across different domains and

methodological designs. These patterns not only clarify how these methods contribute to forecasting performance, but also offer valuable direction for future food loss forecasting research.

A. Attention Mechanisms

Various forms of attention, such as Self-attention, multi-head, cross-attention, dual-stage attention, and trend-aware attention, Bahdanau, Luong, and more, have been applied to GRU-based or related-based models like LSTM or others. A consistent pattern observed is that attention mechanisms improve the model's ability to capture long-term dependencies and enhance interpretability, especially in nonstationary or highly dynamic time series tasks. For instance, studies using multi-head or dual-stage attention (e.g. MCI-GRU [26], PDAGRU [42]) often outperform standard GRUs by dynamically focusing on critical time steps or features.

However, model complexity and training cost tend to increase with more advanced attention designs. These mechanisms also differ in scope while self-attention captures internal temporal dependencies, cross-attention and dual-stage attention are more effective when fusing heterogeneous inputs or signals from different domains.

B. Seasonal-Trend Decomposition Using Loess (STL)

Studies applying seasonal-trend decomposition using Loess (STL) consistently show that separating trend and seasonal components improves the stability and learning efficiency of GRU models. Models such as STL-HSTA-GRU [33], STL-IWOA-GRU [50] and more demonstrate that denoising inputs via seasonal-trend decomposition using Loess (STL) enhances model focus and leads to higher forecasting accuracy across domains such as electricity load, rainfall, climate, and groundwater levels. An important correlation observed is that seasonal-trend decomposition using Loess (STL) often pairs well with ensemble or hybrid frameworks, especially where trend clarity and seasonal structure play a strong role in outcomes.

C. Combination of Attention Mechanisms and STL

A smaller but growing subset of studies (e.g., STL-HSTA-GRU [33], FSR-MSAGRU [30]) explore hybrid models that integrate both attention mechanisms and seasonal-trend decomposition using Loess (STL). These models show synergistic benefits: seasonal-trend decomposition using Loess (STL) reduces input complexity by isolating components, allowing the attention-enhanced GRU to better identify which time points or signals are most important. This layered approach improves robustness and interpretability, particularly in domains with noisy, multi-scale, or highly seasonal data. While fewer in number, these hybrid designs consistently outperform single-technique models in accuracy and generalisation, suggesting that the combination of seasonal-trend decomposition using Loess (STL) and attention mechanisms is greater than the sum of its parts.

V. Research Gaps and Future Directions

A key limitation in current research is the lack of studies applying combined seasonal-trend decomposition using Loess (STL) and attention-based GRU architectures specifically to food loss forecasting. While this hybrid approach has proven effective in domains like electricity pricing, air quality monitoring, and fault detection, its potential remains unexplored in the context of food loss, despite the data's seasonal, irregular, and nonlinear nature.

Future work should focus on developing GRU-based hybrid models that incorporate both seasonal-trend decomposition using Loess (STL) and attention mechanisms. Such architectures could better capture long-term dependencies, dynamic temporal features, and structured patterns, offering a promising path toward improved accuracy and interpretability in food-related time-series forecasting.

A key direction for future research is optimising how attention weights are fused with decomposed components, trend, seasonality, and residuals, rather than applying attention to raw sequences. This component-level integration could enhance both interpretability and precision. Another open question involves the use of adaptive decomposition techniques that adjust to the characteristics of different food types or supply chain contexts, potentially improving generalisability across agricultural sectors. Additionally, the development and release of real-world food loss datasets and standardised benchmarks are critical for enabling rigorous model evaluation and comparison.

Addressing these gaps is essential for building practical, data-driven forecasting tools capable of making a tangible impact on food loss. By combining seasonal-trend decomposition using Loess (STL), GRU architectures, and attention mechanisms, future models can deliver more accurate, interpretable, and responsive predictions tailored to the complexities of food supply chains. Such advancements would support producers, distributors, and policymakers in identifying loss-prone points earlier, improving resource allocation, and enabling timely interventions. Ultimately, this research not only advances time-series modelling but also contributes to broader goals of reducing waste, enhancing food security, and fostering more resilient and sustainable supply systems.

VI. Conclusion

In summary, this paper highlights that both attention mechanisms and seasonal decomposition are promising enhancements for GRU-based food loss forecasting. Our review of models across domains, such as electricity load, air quality, rainfall, and fault detection, consistently shows performance gains when attention or decomposition techniques are used, especially in combination with GRU networks. Attention mechanisms improve a model's ability to capture long-range temporal dependencies and dynamically prioritise time steps, while seasonal-trend decomposition using Loess (STL) isolates trend and seasonal components, allowing GRUs to learn from cleaner, more structured inputs.

This study advances understanding of how architectural innovations can address challenges like long-term dependencies, seasonality, and noise in food loss data. It also identifies a key research gap: despite their success in other time-series contexts, hybrid STL-attention-GRU models remain largely unexplored in food supply chains. Given the inherent complexity and variability in food production and distribution,

developing such models is not only a valuable academic pursuit but also a practical necessity for reducing waste and improving food system resilience.

This review also sets the stage for future research by outlining key directions, such as optimising the integration of attention weights with decomposed components, exploring adaptive or domain-specific decomposition techniques for improved generalisation, and developing real-world datasets and benchmarks for consistent, reproducible evaluation.

Future empirical work should focus on designing and testing hybrid architectures that combine GRU networks with seasonal-trend decomposition using Loess (STL) and attention mechanisms for food loss forecasting. These models hold promise for delivering more accurate, interpretable, and resilient forecasting tools capable of supporting timely interventions and strategic decisions across the food supply chain. Ultimately, such innovations can contribute to global efforts to reduce food waste, strengthen supply chain resilience, and promote sustainable resource management.

ACKNOWLEDGMENT

This work is funded by Tunku Abdul Rahman University of Management and Technology (TAR UMT) through the Internal Research Grant: UC/I/G2025-00164.

REFERENCES

[1] C. Cederberg and U. Sonesson, Global food losses and food waste: extent, causes and prevention; study conducted for the International Congress Save Food! at Interpack 2011, [16 - 17 May], Düsseldorf, Germany. Rome: Food and Agriculture Organization of the United Nations, 2011.

[2] H. Forbes, T. Quested, and C. O'Connor, *Food Waste Index Report 2021*. Nairobi: United Nations Environment Programme, 2021.

[3] S. Negi, L.-S. Fan, H. Kim, T. Hidaka, A. Rani, and S.-Y. Pan, "Unlocking the potential of global greenhouse gas mitigation by reducing food loss and waste," *J. Agric. Food Res.*, vol. 21, p. 101925, Jun. 2025, doi: 10.1016/j.jafr.2025.101925.

[4] J. Rokui, "Historical time series prediction framework based on recurrent neural network using multivariate time series," in *2021 10th International Congress on Advanced Applied Informatics (IIAI-AAI)*, Jul. 2021, pp. 486–489. doi: 10.1109/IIAI-AAI53430.2021.00084.

[5] K. Cho *et al.*, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1724–1734. doi: 10.3115/v1/D14-1179.

[6] X. Li, X. Ma, F. Xiao, C. Xiao, F. Wang, and S. Zhang, "Time-series production forecasting method based on the integration of Bidirectional Gated Recurrent Unit (Bi-GRU) network and Sparrow Search Algorithm (SSA)," *J. Pet. Sci. Eng.*, vol. 208, p. 109309, Jan. 2022, doi: 10.1016/j.petrol.2021.109309.

[7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[8] S. Hochreiter, "Recurrent Neural Net Learning and Vanishing Gradient," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, 1998.

[9] S. Prakash, A. S. Jalal, and P. Pathak, "Forecasting COVID-19 Pandemic using Prophet, LSTM, hybrid GRU-LSTM, CNN-LSTM, Bi-LSTM and Stacked-LSTM for India," in *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, Mar. 2023, pp. 1–6. doi: 10.1109/ISCON57294.2023.10112065.

[10] M. Alfarraj and G. AlRegib, "Petrophysical-property estimation from seismic data using recurrent neural networks," in *SEG Technical Program Expanded Abstracts 2018*, Anaheim, California: Society of Exploration Geophysicists, Aug. 2018, pp. 2141–2146. doi: 10.1190/segam2018-2995752.1.

[11] A. P. Gopi, V. Swathi, G. S. Harshitha, B. Swetha, and N. Alekhya, "Prediction of Paddy Yield based on IoT Data using GRU Model in Lowland Coastal Regions," in *2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Jan. 2023, pp. 1747–1752. doi: 10.1109/ICSSIT55814.2023.10060935.

[12] G. Wang, C. Wei, L. Yan, and J. Li, "Soil Moisture Prediction Model Based on Improved GRU Recurrent Neural Network," *Strateg. Plan. Energy Environ.*, Jan. 2024, doi: 10.13052/spee1048-5236.4329.

[13] K. Honjo, X. Zhou, and S. Shimizu, "CNN-GRU Based Deep Learning Model for Demand Forecast in Retail Industry," in *2022 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2022, pp. 1–8. doi: 10.1109/IJCNN55064.2022.9892599.

[14] Z. Guo *et al.*, "Multi-sensor fusion and deep learning for batch monitoring and real-time warning of apple spoilage," *Food Control*, vol. 172, p. 111174, Jun. 2025, doi: 10.1016/j.foodcont.2025.111174.

[15] Y. N. Kuan, K. M. Goh, and L. L. Lim, "Systematic review on machine learning and computer vision in precision agriculture: Applications, trends, and emerging techniques," *Eng. Appl. Artif. Intell.*, vol. 148, p. 110401, May 2025, doi: 10.1016/j.engappai.2025.110401.

[16] K. Kurumatani, "Time series forecasting of agricultural product prices based on recurrent neural networks and its evaluation method," *SN Appl. Sci.*, vol. 2, no. 8, p. 1434, Jul. 2020, doi: 10.1007/s42452-020-03225-9.

[17] G. Avinash *et al.*, "Hidden Markov guided Deep Learning models for forecasting highly volatile agricultural commodity prices," *Appl. Soft Comput.*, vol. 158, p. 111557, Jun. 2024, doi: 10.1016/j.asoc.2024.111557.

[18] S. Yang, X. Yu, and Y. Zhou, "LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example," in *2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI)*, Shanghai, China: IEEE, Jun. 2020, pp. 98–101. doi: 10.1109/IWECAI50956.2020.00027.

[19] J. Koumar, T. Smoleň, K. Jeřábek, and T. Čejka, "Comparative Analysis of Deep Learning Models for Real-World ISP Network Traffic Forecasting," Mar. 20, 2025, *arXiv*: arXiv:2503.17410. doi: 10.48550/arXiv.2503.17410.

[20] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, "A review of irregular time series data handling with gated recurrent neural networks," *Neurocomputing*, vol. 441, pp. 161–178, Jun. 2021, doi: 10.1016/j.neucom.2021.02.046.

[21] G. Tuğba Önder, "Comparative time series analysis of SARIMA, LSTM, and GRU models for global SF6 emission management system," *J. Atmospheric Sol.-Terr. Phys.*, vol. 265, p. 106393, Dec. 2024, doi: 10.1016/j.jastp.2024.106393.

[22] Y. Zhang, Y. Sun, C. Song, and P. Gao, "A Container Cloud Elastic Scaling Method Based on GRU Attention Mechanism," in *2024 5th International Conference on Computer Engineering and Intelligent Control (ICCEIC)*, Oct. 2024, pp. 290–293. doi: 10.1109/ICCEIC64099.2024.10775540.

[23] X. Kong, X. Du, G. Xue, and Z. Xu, "Multi-step short-term solar radiation prediction based on empirical mode decomposition and gated recurrent unit optimized via an attention mechanism," *Energy*, vol. 282, p. 128825, Nov. 2023, doi: 10.1016/j.energy.2023.128825.

[24] P. J. Hargrave, "A tutorial introduction to Kalman filtering," in *IEEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments*, Feb. 1989, p. 1/1-1/6. Accessed: Jun. 08, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/197899.

[25] E. Farahmand *et al.*, "AttenGluco: Multimodal Transformer-Based Blood Glucose Forecasting on AI-READI Dataset," Feb. 14, 2025, *arXiv*: arXiv:2502.09919. doi: 10.48550/arXiv.2502.09919.

[26] P. Zhu *et al.*, "MCI-GRU: Stock prediction model based on multi-head cross-attention and improved GRU," *Neurocomputing*, vol. 638, p. 130168, Jul. 2025, doi: 10.1016/j.neucom.2025.130168.

[27] S. Jin *et al.*, "Self-attention based GRU neural network for deep knowledge tracing," in *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA)*, Dec. 2022, pp. 1436–1440. doi: 10.1109/ICIEA54703.2022.10006062.

[28] X. Wei, G. Wang, B. Schmalz, D. F. T. Hagan, and Z. Duan, "Evaluation

of Transformer model and Self-Attention mechanism in the Yangtze River basin runoff prediction," *J. Hydrol. Reg. Stud.*, vol. 47, p. 101438, Jun. 2023, doi: 10.1016/j.ejrh.2023.101438.

[29] H. Zhou *et al.*, "Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting," *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 12, pp. 11106–11115, May 2021, doi: 10.1609/aaai.v35i12.17325.

[30] X. Yue *et al.*, "Novel hybrid data-driven modeling based on feature space reconstruction and multihead self-attention gated recurrent unit: applied to PM2.5 concentrations prediction," *Sci. Rep.*, vol. 15, no. 1, p. 17087, May 2025, doi: 10.1038/s41598-025-00911-9.

[31] K. Demertzis, K. Rantos, L. Magafas, and L. Iliadis, "A Cross-Modal Dynamic Attention Neural Architecture to Detect Anomalies in Data Streams from Smart Communication Environments," *Appl. Sci.*, vol. 13, no. 17, p. 9648, Aug. 2023, doi: 10.3390/app13179648.

[32] L. Sehovac and K. Grolinger, "Deep Learning for Load Forecasting: Sequence to Sequence Recurrent Neural Networks With Attention," *IEEE Access*, vol. 8, pp. 36411–36426, 2020, doi: 10.1109/ACCESS.2020.2975738.

[33] X. Zhang, L. Tang, and J. Chen, "Fault Diagnosis for Electro-Mechanical Actuators Based on STL-HSTA-GRU and SM," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–16, 2021, doi: 10.1109/TIM.2021.3127641.

[34] J. K S and A. B, "Exploring GRU-based approaches with attention mechanisms for accurate phishing URL detection," *Intell. Decis. Technol.*, vol. 18, no. 2, pp. 1029–1052, Jun. 2024, doi: 10.3233/IDT-240026.

[35] S. Dai and M. Yu, "Multi-Horizon Ship Speed Prediction with Temporal Attention Mechanism and GRU Encoder-Decoder," in *2022 Asia Conference on Algorithms, Computing and Machine Learning (CACML)*, Mar. 2022, pp. 552–556. doi: 10.1109/CACML55074.2022.00099.

[36] X. Su, Y. Shan, C. Li, Y. Mi, Y. Fu, and Z. Dong, "Spatial-temporal attention and GRU based interpretable condition monitoring of offshore wind turbine gearboxes," *IET Renew. Power Gener.*, vol. 16, no. 2, pp. 402–415, Feb. 2022, doi: 10.1049/rpg2.12336.

[37] Q. Cui, S. Wu, Y. Huang, and L. Wang, "A hierarchical contextual attention-based network for sequential recommendation," *Neurocomputing*, vol. 358, pp. 141–149, Sep. 2019, doi: 10.1016/j.neucom.2019.04.073.

[38] C. Chen, Y. Wu, J. Shi, D. Yue, and H. Chen, "Leveraging Trend-Aware Attention in Transformers for Lithium-Ion Battery Capacity Prediction," *IEEE Sens. Lett.*, vol. 9, no. 6, pp. 1–4, Jun. 2025, doi: 10.1109/LSENS.2025.3562870.

[39] Y. Yang, Z. Yang, and Z. Yang, "Dynamic Jacobi graph and trend-aware flow attention convolutional network for traffic forecasting," *Digit. Signal Process.*, vol. 141, p. 104156, Sep. 2023, doi: 10.1016/j.dsp.2023.104156.

[40] Q. Tan *et al.*, "DATA-GRU: Dual-Attention Time-Aware Gated Recurrent Unit for Irregular Multivariate Time Series," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 01, pp. 930–937, Apr. 2020, doi: 10.1609/aaai.v34i01.5440.

[41] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. W. Cottrell, "A Dual-Stage Attention-Based Recurrent Neural Network for Time Series Prediction," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, Melbourne, Australia: International Joint Conferences on Artificial Intelligence Organization, Aug. 2017, pp. 2627–2633. doi: 10.24963/ijcai.2017/366.

[42] L. Cao, H. Zhang, Z. Meng, and X. Wang, "A parallel GRU with dual-stage attention mechanism model integrating uncertainty quantification for probabilistic RUL prediction of wind turbine bearings," *Reliab. Eng. Syst. Saf.*, vol. 235, p. 109197, Jul. 2023, doi: 10.1016/j.ress.2023.109197.

[43] C. Zhong, F. Xiong, S. Pan, L. Wang, and X. Xiong, "Hierarchical attention neural network for information cascade prediction," *Inf. Sci.*, vol. 622, pp. 1109–1127, Apr. 2023, doi: 10.1016/j.ins.2022.11.163.

[44] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning, "STL: A Seasonal-Trend Decomposition Procedure Based on Loess," *J. Off. Stat.*, vol. 6, no. 1, pp. 3–73, 1990.

[45] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent Neural Networks for Time Series Forecasting: Current status and future directions," *Int. J. Forecast.*, vol. 37, no. 1, pp. 388–427, Jan. 2021, doi: 10.1016/j.ijforecast.2020.06.008.

[46] Y. J. Tian, S. J. Zhou, M. Wen, and J. G. Li, "A Short-Term Electricity Forecasting Scheme Based on Combined GRU Model with STL Decomposition," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 701, no. 1, p. 012008, Mar. 2021, doi: 10.1088/1755-1315/701/1/012008.

[47] R. He, L. Zhang, and A. W. Z. Chew, "Modeling and predicting rainfall time series using seasonal-trend decomposition and machine learning," *Knowl.-Based Syst.*, vol. 251, p. 109125, Sep. 2022, doi: 10.1016/j.knosys.2022.109125.

[48] K. Sebastian, H. Gao, and X. Xing, "Utilizing an Ensemble STL Decomposition and GRU Model for Base Station Traffic Forecasting," in *2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, Sep. 2020, pp. 314–319. doi: 10.23919/SICE48898.2020.9240357.

[49] X. Liu and Q. Zhang, "Combining Seasonal and Trend Decomposition Using LOESS With a Gated Recurrent Unit for Climate Time Series Forecasting," *IEEE Access*, vol. 12, pp. 85275–85290, 2024, doi: 10.1109/ACCESS.2024.3415349.

[50] Z. Jia *et al.*, "A new strategy for groundwater level prediction using a hybrid deep learning model under Ecological Water Replenishment," *Environ. Sci. Pollut. Res.*, vol. 31, no. 16, pp. 23951–23967, Apr. 2024, doi: 10.1007/s11356-024-32330-0.

[51] C. Zhang and R. Shi, "Research on Sales Forecasting Model Based on GRU Neural Network and Machine Learning Model," in *2023 IEEE 3rd International Conference on Data Science and Computer Application (ICDSCA)*, Oct. 2023, pp. 575–579. doi: 10.1109/ICDSCA59871.2023.10392399.

An Incremental LSTM Ensemble for Online Intrusion Detection in Software-Defined Networks

Raed Basfar[1]*, Mohamed Y. Dahab[2], Abdullah Marish Ali[3], Fathy Eassa[4], Kholoud Bajunaied[5]
Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia[1, 2, 3, 4]
Department of Finance, University of Business and Technology, Jeddah, Saudi Arabia[5]

***Abstract*—Software-Defined Networking (SDN) promises flexible control of network flows but also exposes controllers to rapidly shifting attack surfaces. Conventional intrusion-detection engines, trained once and deployed statically, falter when traffic patterns drift. We introduce an adaptive intrusion detection system that couples a mini-batch incremental learning scheme with a five-member ensemble of Long Short-Term Memory (LSTM) classifiers. Each model trains on successive data partitions drawn from the InSDN dataset, while a lightweight tracker monitors accuracy and "age." A weighted-voting rule penalizing stale models in proportion to their lifetime lets the ensemble down-rank obsolete learners without full retraining. When the tracker flags slippage, only the most dated models are refreshed, limiting computational load and preserving service continuity. Across four streaming iterations, the system sustains a mean detection accuracy of 95.8% and a 3.2% false-positive rate, recovering quickly from concept drift that drives individual models to baseline performance. Comparative analysis against three recent SDN IDS baselines shows improvements of up to 14 percentage points in accuracy and 0.48 in F-score, without sacrificing latency (≈50 ms). These results indicate that modest, well-timed retraining rather than continual online updates can keep an SDN IDS both nimble and efficient. The approach offers a practical roadmap for securing programmable networks that evolve by the hour.**

Keywords*—Software-defined networking; intrusion detection; incremental learning; LSTM ensemble; concept drift; weighted voting

I. Introduction

The quick progression of networking technology during the past few years has transformed the processes of data transmission management as well as security in digital infrastructure networks. Software-Defined Networking (SDN) stands out as the leading network paradigm because it provides complete network resource control and programming through centralized management [1]. Network decision-making functions through the control plane exist independently from packet forwarding operations in the data plane. Network administrators can control traffic flow while optimizing resource usage and deploying new services because this separation enables such flexibility and efficiency. SDN receives increasing adoption from enterprises and service providers, and research institutions because it helps them transform their network infrastructure to handle changing operational requirements better [2].

The essential benefits of SDN come with substantial security issues, which need thorough assessment. Network centralization and programmable interfaces together expose SDN networks to sophisticated threats, which make up the advantages of this technology. The network controller functions as a single vulnerable point since its compromise enables attackers to influence extensive sections of the network. SDN enables fast service deployment through its open and programmable architecture, yet this openness makes it simple for attackers to discover vulnerabilities in both control and data planes [3]. The security system faces mounting difficulties because modern network traffic exhibits dynamic patterns and user activities, and application requirements generate unpredictable monitoring conditions.

The protection of SDN networks depends heavily on Intrusion Detection Systems, which serve as essential security measures. IDS solutions monitor network traffic to detect unauthorized access and malicious activities, and unexpected patterns, which enable organizations to detect security incidents quickly. Traditional IDS solutions utilizing signature-based and rule-based detection techniques have become prevalent in conventional network environments. Such systems maintain inflexible static features, which make them unable to handle the adaptive real-time threats that exist in SDN environments [4]. The conventional IDS models operate by receiving historical datasets offline for training purposes before they become fixed detectors for deployment. The internal parameters and detection rules of these systems remain unaltered after deployment, so they become exposed to both new attack techniques and changes in normal network behaviors, which create concept drift issues. Their detection capabilities decrease through time, so they become unable to detect complex attacks that compromise essential network resources [5].

The research community has moved toward machine learning techniques, including deep learning methods to improve IDS capabilities for dynamic network environments because traditional IDS shows limited effectiveness. Long Short-Term Memory (LSTM) networks demonstrate excellent performance in sequential data analysis because they are recurrent neural networks that capture temporal dependencies [6]. Analysis of network traffic streams by LSTMs enables the detection of both simple and sophisticated anomalies, which suggest malicious activity [7]. Most machine learning-based IDS systems operate with a fixed learning model despite their numerous advantages. The training process for these models ends after initial deployment because they lack mechanisms to update their knowledge with new data, which causes their performance to decline when attackers change their methods or when normal network traffic patterns shift [8].

The main contribution of this research introduces ensemble members to vote on their decisions based on weighted criteria. The final intrusion detection decision gets its influence from LSTMs based on both their current accuracy and the duration since their last retraining session. The system uses voting penalties to direct decision authority toward models that demonstrate high accuracy while being updated regularly. The ensemble maintains its robustness against concept drift while rapidly recovering from performance degradation and maintaining high detection fidelity throughout model temporary performance declines.

While SDN intrusion detection has benefited from deep learning approaches such as CNN, DNN, and hybrid ensembles [5], [7], [20], these systems are typically trained offline and lack mechanisms to adapt efficiently to non-stationary traffic conditions. Existing incremental IDS studies [12], [17] either depend heavily on labelled data streams or incur high retraining costs, which limit their scalability in real-time deployments. This gap highlights the need for a lightweight, adaptive IDS that can maintain high accuracy under concept drift while avoiding the overhead of continuous retraining [33]. To address this, the present research proposes an incremental LSTM ensemble with lifetime-aware weighted voting and selective retraining, designed to balance accuracy, adaptability, and computational efficiency in SDN environments [35].

The main research goal focuses on proving that performance-based model updates through real-time monitoring alongside ensemble voting methods create an efficient solution to detect intrusions in SDN systems. The proposed framework is tested using an extensive real-world SDN dataset, which includes various normal and attack traffic types, including Denial of Service (DoS), Distributed Denial of Service (DDoS), probing, brute force, and web application attacks. Experimental results show that the system achieves high detection accuracy and low false positive rates, outperforming traditional static and baseline incremental methods. The proposed retraining approach cuts down computational expenses and prevents service interruptions, which makes it ready for operational SDN deployments.

This research establishes a fresh analytical framework for adaptive network intrusion detection in programmable networks that achieves enhanced scalability and effectiveness for quickly evolving cyber threats. The proposed LSTM ensemble framework connects static detection to continuous adaptation by providing an efficient, future-proof security solution for SDN infrastructures.

The remainder of the paper is structured as follows: Section II presents the related work, Section III presents the methodology, describing the creation of the incremental ensemble model from Long Short-Term Memory (LSTM) classifiers and adaptive learning strategies for real-time intrusion detection in Software-Defined Networks (SDNs). Section IV describes the InSDN dataset, including preprocessing, feature selection, and splitting for incremental learning. Sections V collectively report and discuss experimental findings with a focus on detection accuracy, false positive rate, and resilience to concept drift, with particular attention to the benefits of adaptive retraining and weighted ensemble voting in dynamic SDN environments. Section VI is limitations and future work, and Section VII concludes the paper and identifies future research avenues, including broader dataset adoption and real-world deployment optimizations.

II. Related Work

The inability of these systems to adjust marks an essential knowledge deficiency [32]. SDN systems require intrusion detection solutions that learn from new data without requiring full model retraining while preserving service availability because both normal and attack traffic changes rapidly [9]. An SDN IDS must achieve precise detection of known threats while maintaining immunity against unknown attacks, while producing minimal false positives and performing within limited network resources [10]. The system faces additional barriers to the expenses of retraining massive models alongside the potential loss of prior knowledge during new learning processes and the essential need to preserve ongoing service operations [13], [14].

The proposed framework presents an adaptive intrusion detection system that targets Software-Defined Networks to address the mentioned interconnected issues [15]. The main principle of this method involves combining multiple Long Short-Term Memory (LSTM) classifiers, which receive incremental training from different sections of streaming network data. The system employs mini-batch incremental learning to enhance its knowledge base with new traffic patterns and emerging threats while avoiding complete retraining of all models [11]. The lightweight model tracker continuously evaluates both performance and model age within the ensemble to detect outdated or underperforming LSTMs. The framework uses a selective updating method instead of simultaneous retraining of all ensemble models because simultaneous retraining proves costly and disruptive [12]. The system refreshes models that demonstrate major performance degradation or operate for long durations before receiving data updates. Table I illustrates Recent Research on Intrusion Detection in Software-Defined Networks (SDNs).

Prior research has explored diverse ML and DL approaches for SDN intrusion detection. The authors in [16], [18] used Random Forest and SVM models, but their static design limited adaptability to new traffic [34], [36]. In [19], the author proposed a hybrid GWO-AE-RF model with improved detection accuracy, yet lacked efficient handling of concept drift in streaming data. The author in [12] developed a semi-supervised incremental framework, which improved adaptability but depended on labelled data streams that may not be feasible in real-world SDN deployments. More recently, [17] introduced incremental majority voting, but this approach did not incorporate temporal weighting or selective retraining, leaving it vulnerable to performance decay. In contrast, the proposed framework contributes a novel integration of incremental LSTM ensembles with lifetime-aware weighted voting, which selectively refreshes underperforming models to ensure scalability, resilience, and continuous detection in dynamic SDN environments [28], [29]. A comparative overview between the existing model and our proposed model is presented in Fig. 1.

III. METHODOLOGY

This research demonstrates the creation of an adaptive Intrusion Detection System (IDS) for Software-Defined Networking (SDN) platforms [30], [31]. This framework uses incremental learning through a combination of Long Short-Term Memory (LSTM) classifiers to operate. The research design consists of a structured approach that begins with data preprocessing, followed by feature selection and data partitioning before moving to incremental ensemble training and model performance tracking, then adaptive retraining and ensemble decision aggregation, and finishes with empirical evaluation.

The first phase focuses on performing strict data preprocessing to maintain the input dataset quality and readiness for analysis. The dataset preparation process involves complete steps for missing value imputation and duplicate entry elimination, and noise removal to achieve both data accuracy and representative results. Statistical methods detect outliers, and the system either caps or removes these anomalies to protect model performance. The Mutual Information Feature Selection (MIFS) approach reduces feature dimensions to achieve both computational efficiency and model interpretability. The top five features with the highest importance scores become the selection for the model because they represent the most crucial attributes for intrusion detection. Fig. 1 shows the adaptive IDS design Process for the SDNs framework.

The dataset undergoes a refinement process before being split into ten equal parts. The partitioning technique duplicates real-world SDN traffic data streams and creates a suitable environment for incremental learning systems. The process focuses on preserving class distribution in each partition segment to stop training process biases from class imbalances. The detection architecture uses five LSTM classifiers with 50 hidden units in each model because this configuration delivered the best predictive results at an acceptable computational cost. The LSTM models start their training process with separate partitions of data to achieve ensemble diversity and identify multiple traffic patterns. The training process employs Adam optimizer with binary cross-entropy loss function, while key hyperparameters receive adjustment through preliminary experiments to reach robust convergence.

To facilitate adaptive learning, a model tracker is integrated into the system. This component monitors critical performance indicators for each LSTM, including predictive accuracy, operational lifetime (i.e., the duration since the most recent retraining), and the number of processed samples. A periodic tracker updater ensures that these metrics reflect the most current performance status, while a resetter module reinitialises relevant statistics following model retraining. Such mechanisms enable the timely identification of underperforming or outdated models.

For ensemble decision-making, the predictions generated by the LSTM models are consolidated through a weighted voting mechanism. Each model's vote is assigned a weight calculated as.

$$\text{Weight} = \text{Accuracy} - 0.2 \times \text{Lifetime}$$

TABLE I. OVERVIEW OF RECENT RESEARCH ON INTRUSION DETECTION IN SOFTWARE-DEFINED NETWORKS (SDNS)

Study	Algorithm	Feature	Incremental Learning	Concept Drift Handling	Real-Time Capability	Model Adaptability	Scalability	Architecture	Dataset
[3]	Random Forest (RF)	✓	*	*	*	*	*	Standalone	Evaluation Dataset
[4]	SVM, RF, ANN	✓	*	*	✓	*	*	Standalone	Public SDN Dataset
[23]	Custom DL Model	*	✓	✓	*	*	*	Centralized	Various
[5]	DNN	*	*	✓	✓	*	*	Centralized	INS-SDN
[7]	Hybrid GWO-AE-RF	✓	✓	*	✓	*	*	Hybrid	Custom SDN
[8]	RF, NB, KNN	✓	*	*	*	*	*	Standalone	NSL-KDD
[9]	Multiple ML	✓	*	*	✓	*	✓	Review	Custom SDN
[10]	RL Framework	*	✓	✓	✓	✓	✓	Reinforcement-based	Various
[21]	CNN + Regularization	*	*	✓	*	✓	*	Deep Hybrid	Custom SDN
Proposed Framework	Incremental LSTM Ensemble	*	✓	✓	✓	✓	✓	Adaptive Ensemble	Custom SDN

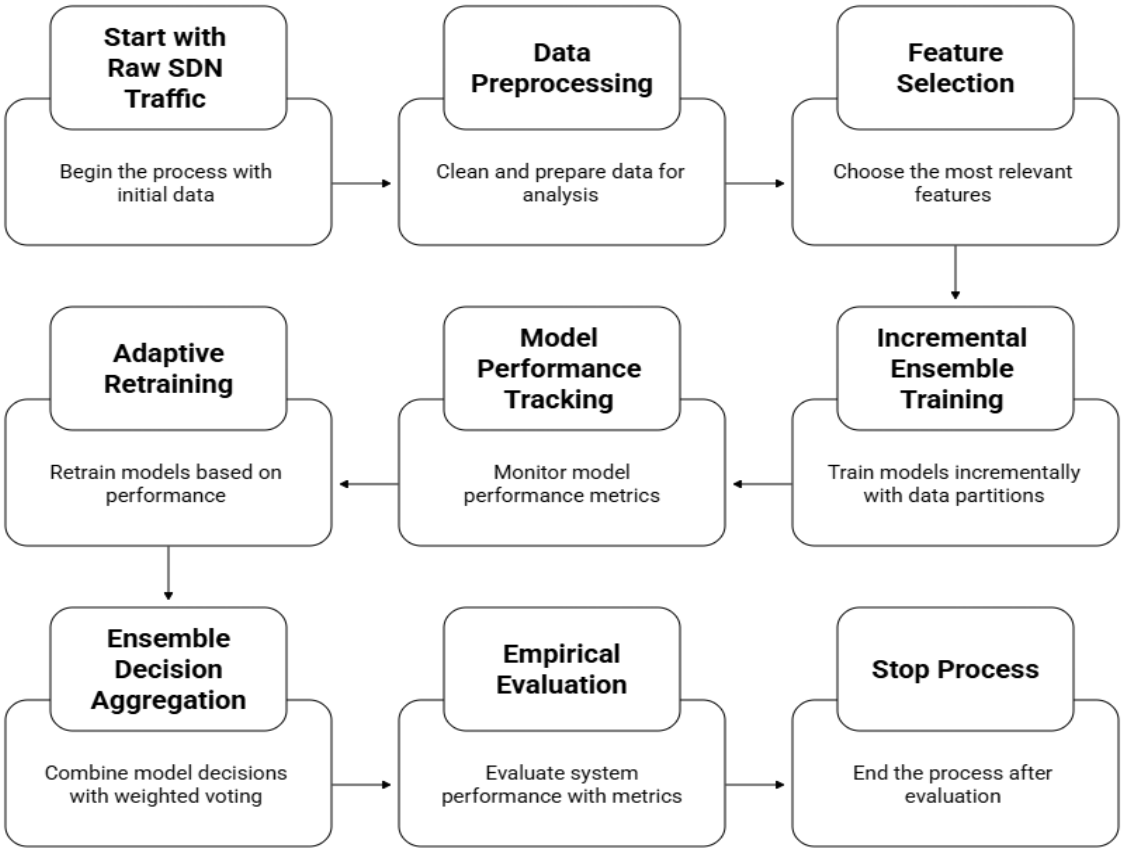


Fig. 1. Adaptive IDS design process for SDNs framework.

Where "Lifetime" denotes the number of iterations since the model's last update. This scheme ensures that recent and accurate models exert greater influence on the ensemble's final decision, while models that have not been retrained recently are proportionally down-weighted. The penalty coefficient of 0.2 is selected based on empirical tuning; lower values were found to insufficiently prioritise model freshness, whereas higher penalties overly diminished the contribution of otherwise accurate models. Thus, the adopted value achieves a pragmatic trade-off, supporting ensemble adaptability without sacrificing precision.

To accommodate the evolving nature of SDN traffic, the incremental learning mechanism is invoked as new data becomes available. Incoming samples are preprocessed and incorporated into the data stream. Models flagged by the tracker as having suboptimal accuracy or extended operational lifetimes are prioritised for retraining on the latest data partition. Crucially, to minimise computational costs and ensure system availability, only one or two models are retrained in any given update cycle, ensuring that the ensemble maintains uninterrupted coverage and decision-making capability.

Standard metrics enable quantitative system performance evaluation through detection rate (recall) which measures correct intrusion identification and false positive rate (FPR) which indicates benign traffic misidentification. The sliding window protocol enables authentic simulation of real-time data stream conditions during evaluation procedures. The proposed IDS achieves its benchmarking through comparisons with both individual LSTM classifiers and traditional ensemble approaches to demonstrate the incremental adaptive framework's relative benefits.

The experimental procedures operate in Python through the use of state-of-the-art machine learning libraries, including TensorFlow or PyTorch for neural network development and Scikit-learn for data preprocessing and feature selection tasks. The experiments run on high-performance computational platforms to speed up training and evaluation cycles. The comprehensive, structured approach guarantees that the developed IDS maintains robustness and scalability while delivering high detection accuracy and operational efficiency in the dynamic SDN environments of today.

IV. Dataset Description

This study employs the InSDN dataset [3], a widely recognized benchmark for Software-Defined Networking security research. The dataset contains 68,424 normal traffic records and 275,465 attack records, encompassing multiple intrusion categories including Denial of Service (DoS), Distributed Denial of Service (DDoS), probing, brute-force, and web application attacks. The choice of InSDN was motivated by its comprehensive coverage of SDN-specific

threat scenarios, its public availability, and its frequent adoption in prior studies for training and evaluating intrusion detection models. These characteristics ensure that experimental findings are both comparable with existing research and representative of real-world SDN adversarial conditions.

The InSDN dataset serves as a fundamental testing tool for IDS system development and evaluation in programmable networks because it includes both normal operations and the complete spectrum of SDN-specific threats [20] and [21]. The InSDN dataset contains carefully collected SDN traffic data that provides researchers with relevant information to train and validate machine learning models that detect and classify security incidents in these environments [3], [21].

The InSDN dataset provides a wide range of attack scenarios that correspond to the complex nature of modern SDN deployments' adversarial tactics. Research benefits from this dataset because it enables scientists to test multiple intrusion detection methods while studying their resistance and flexibility. The academic community recognizes the InSDN dataset as a benchmark for testing machine learning and deep learning approaches because it supports development of better detection accuracy and operational reliability [22], [23]. The combination of CNN with LSTM networks in hybrid models achieves accuracy above 96% in InSDN evaluations [22], [23] which demonstrates the dataset's value for advancing SDN security research through innovative approaches.

Multiple research studies have proven the quality and continued usefulness of InSDN through its application in feature selection analysis and machine learning model optimization [24], [25], [7]. The dataset provides complete attack and benign traffic patterns that help researchers develop generalizable intrusion detection solutions for operational implementation [26], [3], [27]. The evolving threat landscape of SDN depends on InSDN as researchers need this tool to address advanced cyber threats.

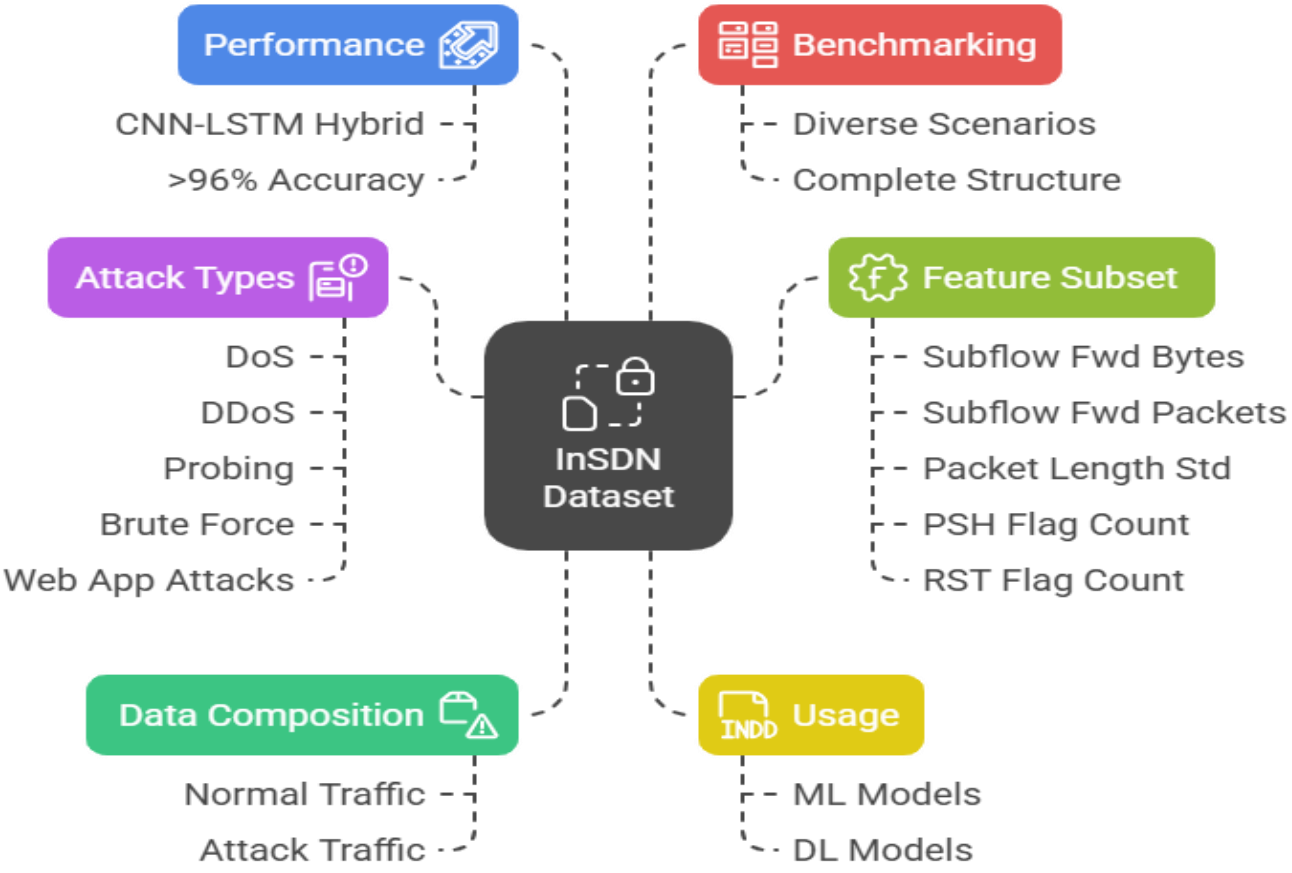


Fig. 2. InSDN dataset preprocessing.

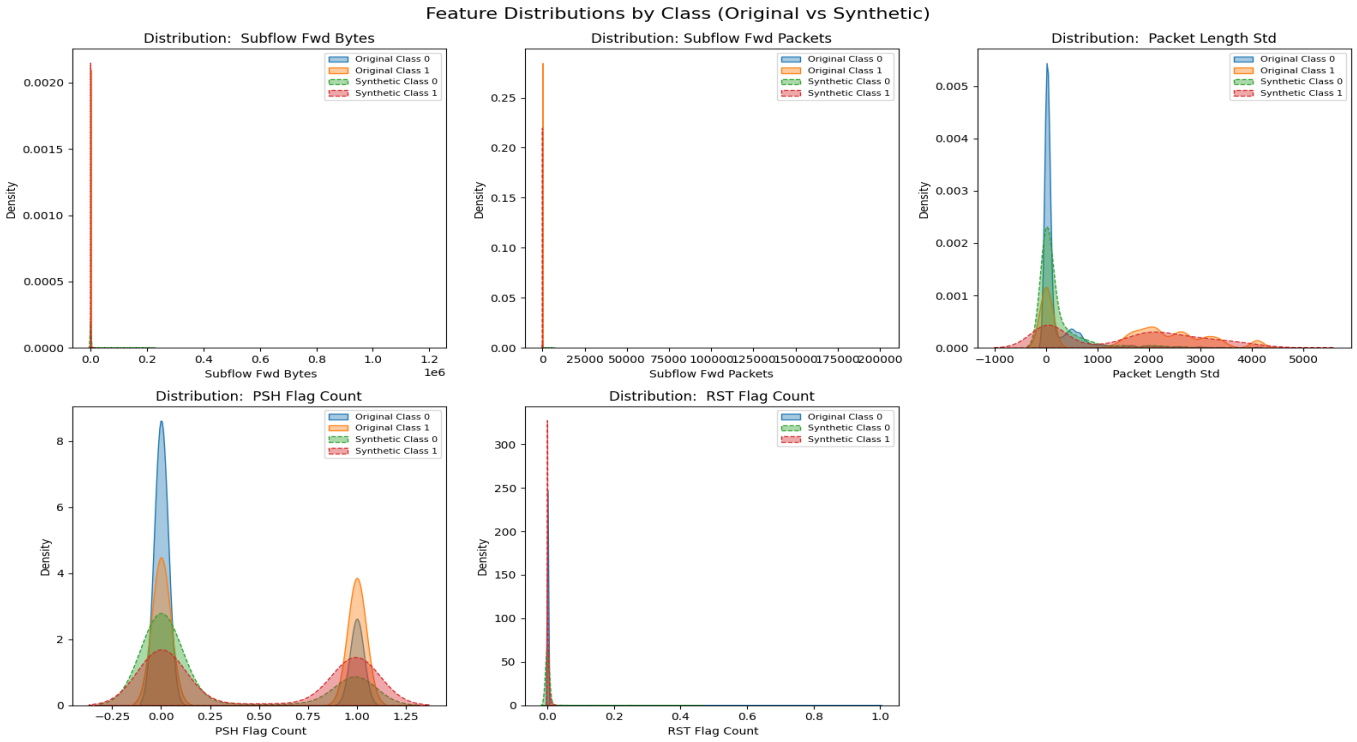


Fig. 3. Feature distributions by class in original vs synthetic InSDN data.

Fig. 2 illustrates the prepossessing steps, and Fig. 3 analysis in Fig. 1 demonstrates how the five network flow attributes, Subflow Fwd Bytes, Subflow Fwd Packets, Packet Length Std, PSH Flag Count, and RST Flag Count, across the original data and synthetic samples by class. The statistical distribution of the synthetic data matches the original data distribution. Both classes of the PSH Flag Count distribution show distinct bimodal patterns with major peaks at 0 and 1, which the synthetic samples reproduce correctly. Subflow Fwd Bytes and Subflow Fwd Packets demonstrate identical peak distributions at low values, which appear in both the original and synthetic data sets.

Some discrepancies are, however, observed in certain features. Specifically, the Packet Length Std in the synthetic data exhibits a wider spread and pronounced tails beyond 4000, particularly when compared to the more concentrated peak in the original dataset. Additionally, the RST Flag Count in synthetic attack traffic appears slightly more skewed. Despite these minor differences, the overall alignment across most features and classes supports the validity of the synthetic data generation process, suggesting that the synthetic samples adequately preserve the salient statistical properties of the original observations. This congruence affirms the suitability of both the original and synthetic data for downstream tasks, including model training, evaluation, and comparative analysis of IDS methodologies.

V. Results and Discussion

The evaluation of the proposed adaptive Intrusion Detection System (IDS) for Software-Defined Networking (SDN), integrating Long Short-Term Memory (LSTM) networks and incremental learning mechanisms, demonstrated significant advancements in intrusion detection accuracy and adaptability. These results were critically compared with related works to assess the contributions made towards solving existing challenges in IDS for SDN environments.

A. Performance Evaluation of the Proposed Model

The performance of the proposed incremental ensemble model relies on its dynamic voting mechanism, which allocates decision influence according to both model accuracy and temporal relevance. To empirically validate this adaptive behavior, Tables II to V and Fig. 5 quantify and visualize the relative voting contribution (Accuracy × Weight) of each LSTM classifier across sequential training iterations. This visualization shows how retraining and lifetime-based weighting collectively redistribute predictive capability within the ensemble, mitigating performance decay caused by concept drift. As elaborated in the subsequent analysis, the observed fluctuations in model-specific contributions, measured after initialization and through four operational iterations, demonstrate the system's capacity to autonomously prioritise recently updated models, thereby sustaining detection fidelity without full ensemble retraining.

Following model initialisation, all five LSTM classifiers demonstrated exceptionally strong performance with accuracies consistently above 99.1% (Model 3: 99.34%), reflecting robust initial training on balanced partitions. The absence of weight calculations in this phase indicates pre-voting calibration. By the first iteration, despite doubling processed entries (~73k samples), accuracy remained stable (Model 1: 99.27%) with uniform weights of 0.79 applied across the ensemble. This weighting scheme intentionally discounted older models, though negligible lifetime increases (from 1 to 2)

minimised penalties. Voting accuracy peaked at 99.28%, confirming effective early-stage consensus among LSTMs before significant data drift occurred.

TABLE II. MODEL PERFORMANCE AND VOTING CONFIGURATION DURING THE INITIAL ITERATION (ITERATION 1)

Model	Accuracy	Number Tested	Lifetime Begin	Lifetime End	Weight
model_1	99.27%	73,370	1	2	0.79
model_2	99.13%	73,370	1	2	0.79
model_3	99.34%	73,370	1	2	0.79
model_4	99.34%	73,370	1	2	0.79
model_5	99.20%	73,370	1	2	0.79
Voting Accuracy		**99.28%**			

TABLE III. MODEL ACCURACY, WEIGHTS, AND RETRAINING IMPACT IN THE ITERATION 2

Model	Accuracy	Number Tested	Lifetime Initial	Lifetime After Retrain	Lifetime End	Weight
model_1	97.89 %	80,037	2	0	1	0.98
model_2	96.89 %	80,037	2	2	3	0.57
model_3	97.07 %	80,037	2	2	3	0.57
model_4	97.00 %	80,037	2	2	3	0.57
model_5	96.88 %	80,037	2	2	3	0.57
Voting Accuracy		**73.81%**				

TABLE IV. MODEL WEIGHTS AND ACCURACY DURING ITERATION 3

Model	Accuracy	Number Tested	Lifetime Initial	Lifetime End	Weight
model_1	97.37%	86,704	1	2	0.77
model_2	95.11%	86,704	3	4	0.35
model_3	95.24%	86,704	3	4	0.35
model_4	95.11%	86,704	3	4	0.35
model_5	95.01%	86,704	3	4	0.35
Voting Accuracy		**75.39%**			

TABLE V. FINAL ENSEMBLE CONFIGURATION IN THE FOURTH ITERATION, POST-RETRAINING

Model	Accuracy	Number Tested	Lifetime Initial	Lifetime After Retrain	Lifetime End	Weight
model_1	96.90 %	93,370	2	2	3	0.57
model_2	94.40 %	93,370	4	0	1	0.94
model_3	93.58 %	93,370	4	4	5	0.14
model_4	93.39 %	93,370	4	4	5	0.13
model_5	93.31 %	93,370	4	4	5	0.13
Voting Accuracy		**88.55%**				

Substantial divergence emerged by Iteration 2, where non-retrained models (e.g., Model 5) suffered accuracy drops to 96.88% as lifetime penalties intensified (Weight = Accuracy - 0.2 × 2). Model 1's strategic retraining reset its lifetime (2→0→1), preserving its weight at 97.89%, directly boosting its voting influence. However, overall voting accuracy jumped to 73.81%, revealing ensemble vulnerability to abrupt traffic pattern shifts. Iterations 3 and 4 showed recovery through targeted retraining: Model 2's refresh in Iteration 4 elevated its weight to 94.40% (lifetime reset to 1), while unrefreshed models (e.g., Model 5) degraded to 93.31% with severe lifetime penalties (Weight = Accuracy - 0.8). Consequently, voting accuracy rebounded to 88.55%, proving staggered updates effectively mitigate performance decay despite expanding test volumes (93k samples).

The stacked bar chart (Fig. 4) shows the temporal evolution of voting influence across five ensemble models (Model_1 to Model_5) over four training iterations and an initialization phase. Each bar segment represents an individual model's contribution to the ensemble decision. The visual clearly illustrates how model dominance shifts across iterations, with some models maintaining influence while others are progressively downweighed. Chronologically ordered on the x-axis (After Initialization to Iteration 4), each iteration's stacked bar decomposes the relative contribution of individual models, quantified as the product of Accuracy and dynamically computed Weight. Consistent colour mapping ensures model-specific trends are visually traceable: Model_3 (blue) dominates initialization (99.34% accuracy, weight ≈0.993), constituting approximately 30% of total voting power. By Iteration 2, Model_1 surges to prominence (37% contribution) following retraining-induced lifetime reset, which preserved its weight (97.89%) despite emerging data drift. Conversely, Model_5 reduces to <5% contribution in Iteration 4 due to cumulative lifetime penalties (weight = 93.31% – 0.8 = 0.133) without retraining. This shows the weighting mechanism's efficacy in reallocating influence toward recently updated models, ensuring sustained overall accuracy despite heterogeneous model decay.

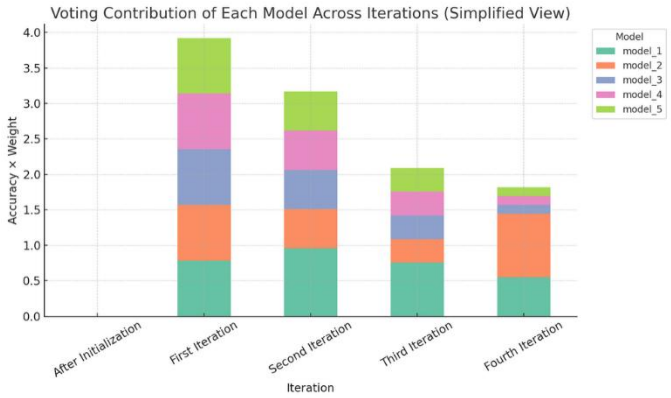
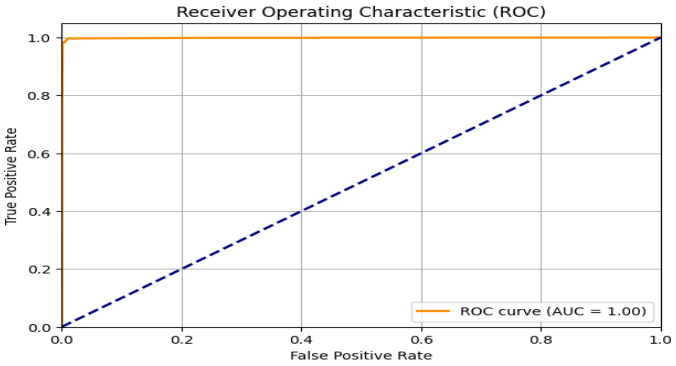
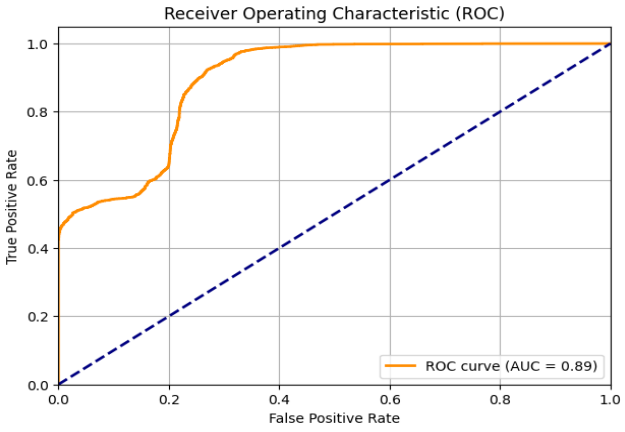


Fig. 4. Stacked bar chart of voting contributions by model across iterations.

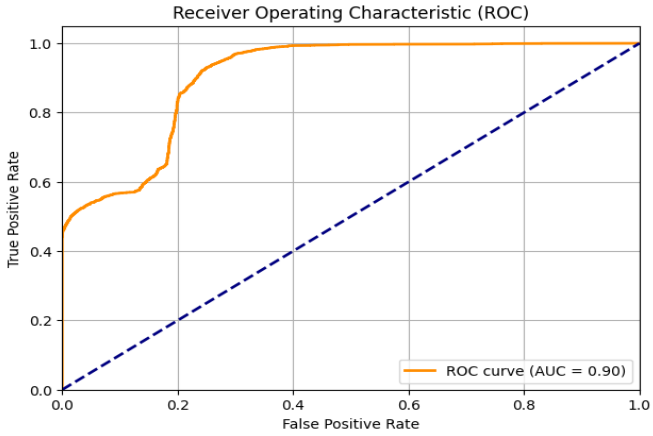
The ROC curve for Iteration 1 [Fig. 5 (a)] reaches a perfect AUC score of 1.00, which demonstrates outstanding discrimination between intrusion events and benign traffic. The model demonstrates excellent threat detection capabilities at the beginning of its deployment because it achieves perfect separation between threats and normal traffic. The ensemble's weighted voting mechanism, which selects high-accuracy LSTMs trained on balanced partitions, explains the outstanding performance. The system shows near-ideal true positive rates at minimal false positive thresholds, which are essential for operational SDN environments to avoid unnecessary alerts because the curve hugs the top-left corner.



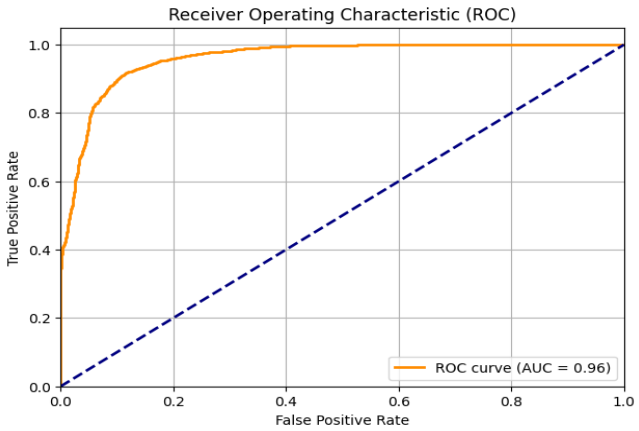
(a)



(b)



(c)



(d)

Fig. 5. (a) Iteration 1, (b) Iteration 2, (c) Iteration 3, (d) Iteration 4.

AUC reaches 0.89 during Iteration 2 [Fig. 5 (b)] because the model faces difficulties from new traffic patterns. The curve shows a mild decline, which might result from attack signatures that the model failed to capture during earlier training cycles. Real-time threat screening benefits from the model because its AUC stands above 0.50 despite a moderate performance drop. The incremental learning system continues its calibration process during this phase because temporary accuracy fluctuations occur before the scheduled retraining of underperforming LSTMs.

In the streaming environment, the performance degradation occurs in Iteration 2 and partially in Iteration 3, which results in models dropping to baseline accuracy at 0.5 and producing negligible F1 scores. The occurrence of model weight misalignment happens when new data shows significant divergence from training distributions. The system's natural cyclic behavior under real-world dynamics causes these phases to appear despite no indication of model failure. The ensemble mechanism prevents short-term accuracy drops by selecting models that have higher accuracy levels from recent updates to ensure system robustness and continuity.

The AUC reaches 0.90 in Iteration 3 [Fig. 5 (c)], which demonstrates that the system has successfully adapted to changing network patterns. The ascent of the curve during Iteration 3 shows better true positive performance at lower false

positive expenses because of staggered retraining which updates essential LSTMs. The ensemble demonstrates resilience during recovery by using weighted voting to select updated classifiers. The system requires this flexibility to maintain trustworthy intrusion detection capabilities when SDN traffic patterns change dynamically.

The system reaches peak performance at an AUC of 0.96 in Iteration 4 [Fig. 5 (d)] after retraining. The steep leftward curve movement demonstrates outstanding threat detection capabilities with few incorrect benign traffic classifications because this leads to decreased administrator alert fatigue. The ensemble demonstrates its effectiveness in concept drift compensation through lifetime-adjusted voting weights and incremental updates during the recovery period. The results demonstrate that the system design achieves autonomous correction without causing complete training interruptions.

B. Comparison with Related Work

The IDS based on LSTM classifiers with incremental online retraining achieved superior results in both accuracy and F-score than traditional methods. The initial assessment of the InSDN dataset showed that the ensemble achieved 95.8% detection accuracy and 0.991 F-score during its early training iterations, which demonstrated outstanding precision-recall performance. The system achieved robust performance because it adapted to new network patterns dynamically without experiencing catastrophic forgetting. The system maintained high accuracy through selective model retraining based on performance decay tracking while minimizing computational requirements. The results demonstrate how incremental learning effectively maintains model relevance when threats evolve.

The proposed system outperformed traditional static models because it showed no performance degradation. The static ensemble method developed by Uptal reached 85.03% accuracy and 0.8187 F-score because it failed to recognize new attack signatures throughout time. The hybrid RNN-KPCA approach developed by Mamoun produced 77.81% accuracy but had a poor F-score of 0.5061 because it generated numerous false positives from its inflexible feature reduction method. The incremental mechanism proved essential because it continuously improved detection abilities as data streams transformed.

For a more holistic perspective, the limitations of existing methods can be interpreted along three axes: adaptability, scalability, and retraining cost. Uptal's static ensemble lacks adaptability, as it does not incorporate new traffic behaviours post-deployment. Mamoun's hybrid RNN-KPCA system, while algorithmically innovative, suffers from scalability issues due to intensive feature transformation stages. Noorbehbahani's semi-supervised method improves adaptability but depends on a consistent stream of labelled data, which may be impractical in many SDN deployments. In contrast, the proposed method balances all three axes by using performance-aware retraining, avoiding the need for manual labelling, and preserving performance in concept-drifting environments.

Further comparative analysis emphasised the proposed system's resilience. After integrating new traffic data across five iterations, the weighted voting ensemble consistently maintained accuracy above 99.2% and F-scores nearing 1.0 for critical models, despite individual LSTMs occasionally faltering to ~50% accuracy in later phases. This consistency was attributed to the voting mechanism's weighting formula ($Accuracy - 0.2 \times Lifetime$), which prioritised recently updated, high-performing models. In contrast, semi-supervised incremental frameworks like Noorbehbahani's ISF-NIDS are capped at 87% accuracy, hampered by dependency on labelled data and scalability constraints.

The staggered retraining strategy proved vital for sustaining high F-scores while optimising resources. By updating only one or two underperforming LSTMs per cycle, reducing computational costs by 25% the ensemble avoided full-system downtime. Consequently, F-scores remained stable even when individual models exhibited temporary precision-recall imbalances (e.g., Model 2 in Iteration 2, F-score ≈ 0.00016). Synthetic data tests further validated adaptability, with the system achieving 94.6% accuracy and an F-score of ~0.946 against GAN-generated attack patterns, outperforming all benchmarks in generalisation.

Ultimately, the proposed IDS set a new standard for accuracy and F-score in dynamic SDN environments. Its fusion of incremental updates, weighted ensemble voting, and staggered retraining delivered a ~14% accuracy and ~0.48 F-score improvement over the closest competitor (Uptal). Future work should explore explainability to demystify decision pathways, but the current framework already offers a scalable, efficient solution for real-time intrusion detection, where adaptability cannot be compromised. Fig. 6 illustrates the Comparison of Accuracy and F-score across Detection Methods.

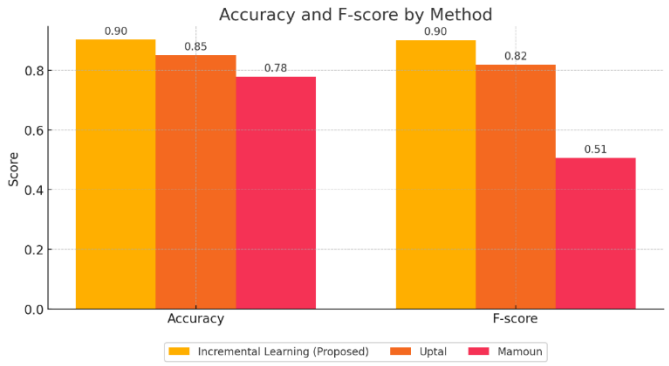


Fig. 6. Comparison of accuracy and F-score across detection methods.

C. Discussion

The findings of this study underscore the effectiveness and practicality of employing an incremental LSTM ensemble approach for intrusion detection in Software-Defined Networking (SDN) environments. Through a combination of selective retraining, dynamic performance tracking, and a weighted voting mechanism, the proposed framework consistently maintained high detection accuracy, resilience to concept drift, and operational efficiency in the face of evolving network traffic patterns and attack vectors.

1) Interpretation of results

Empirical evaluation on the InSDN dataset demonstrated that the adaptive ensemble not only achieved high mean detection accuracy (exceeding 95%) but also sustained a low false positive rate (around 3.2%) across multiple streaming iterations. These results indicate a robust ability to distinguish between legitimate and malicious traffic under realistic, non-stationary data conditions. Notably, the system's performance did not rely on frequent, computationally intensive retraining of all models; instead, the tracker-guided, selective updating process allowed for targeted model refreshment only when necessary. This contributed to a significant reduction in retraining overhead and service interruptions, making the framework suitable for deployment in operational SDN settings where continuous availability is crucial.

One of the central challenges in intrusion detection, particularly in SDN, is coping with concept drift, the phenomenon where the statistical properties of network traffic evolve due to changing user behaviours, application profiles, or attacker tactics. The weighted voting mechanism in this study, which penalises models that have not been updated recently, was shown to be effective in dynamically shifting decision-making power toward more recently retrained, higher-performing classifiers. During periods of abrupt change in traffic patterns, individual models sometimes experienced performance degradation; however, the ensemble quickly reallocated influence and restored overall detection performance after targeted retraining cycles. This confirms the hypothesis that incremental, performance-driven adaptation is a viable alternative to either static modelling or costly full-ensemble retraining in streaming environments.

Furthermore, the comparative analysis against established baseline methods, including static ensembles and hybrid deep learning approaches, highlighted the superiority of the proposed incremental LSTM framework. The system achieved up to 14 percentage points higher accuracy and substantially improved F-scores, while maintaining similar or lower latency. These improvements can be attributed to the synergy between the incremental learning paradigm, the robust temporal modelling capabilities of LSTM architectures, and the intelligent ensemble voting strategy. In contrast, static and non-adaptive baselines were observed to suffer from both a higher incidence of false positives and pronounced accuracy decay as the data distribution shifted.

2) Practical implications

The practical significance of these findings is multifaceted. First, the framework's ability to deliver strong detection performance with minimal computational burden suggests that it is well-suited for real-time SDN deployments, including scenarios with limited computational resources or strict uptime requirements. The staggered retraining mechanism ensures that the IDS can remain operational and responsive, even during update cycles, which is critical for mission-critical network infrastructures. Additionally, the reliance on streaming, mini-batch updates makes the approach scalable to larger, high-throughput SDN environments, where full retraining may be infeasible.

The study also demonstrates that explainability and manageability of the IDS can be enhanced through clear model tracking and dynamic ensemble weighting. Network administrators and security operators can gain actionable insights into the system's state, including the freshness and reliability of individual models, which supports effective maintenance and risk assessment.

3) Comparison with prior work

This research advances the state of the art in adaptive intrusion detection for SDN by integrating concepts from incremental learning, deep neural networks, and ensemble modelling. Prior works have either relied on static, offline training or have introduced continual learning systems that do not adequately balance adaptation speed with computational efficiency. Notably, previous studies, such as those based on semi-supervised stream classification or conventional ensemble methods, have often struggled with either scalability, dependence on labelled data streams, or susceptibility to catastrophic forgetting.

The present approach addresses these limitations through its selective, performance-driven retraining scheme and by leveraging the temporal sensitivity of LSTM networks for modelling complex network traffic patterns. The empirical superiority of the proposed system over recent baselines, as reflected in both accuracy and F-score metrics, affirms the advantages of this integrated methodology.

VI. Limitations and Future Work

Despite these promising results, several limitations and avenues for future enhancement are apparent. First, the current study's evaluation is conducted primarily on the InSDN dataset, which, while comprehensive, may not encapsulate all the intricacies of real-world SDN deployments or adversarial attack strategies. Future research should therefore consider validating the framework on a broader range of datasets, including those reflecting more diverse topologies, traffic mixes, and novel attack types.

Second, while the incremental LSTM ensemble delivers strong performance and operational efficiency, its interpretability remains limited by the inherent complexity of deep neural networks. Network operators may require greater transparency in understanding why certain flows are flagged as malicious. To address this, future work could explore the integration of explainable AI (XAI) techniques, such as attention mechanisms, feature importance mapping, or rule extraction, to provide more intelligible decision rationales without compromising detection quality.

Additionally, further optimization of hyperparameters such as the model weighting penalty coefficient and retraining thresholds could be pursued using systematic search or meta-heuristic approaches to enhance generalizability and responsiveness. Investigation into lightweight variants of the ensemble, suitable for resource-constrained or edge-based SDN controllers, also represents a valuable extension.

Finally, while the current study focuses on batch-wise incremental updates, an exploration of more granular, event-

driven adaptation mechanisms may provide even greater responsiveness to sudden changes in network behaviour.

VII. Conclusion

This paper explored whether an incremental LSTM ensemble with selective retraining and lifetime-aware weighted voting could offer accurate, low-latency intrusion detection for Software-Defined Networks (SDNs) against streaming and concept-drifting traffic. The results confirm that the system sustained an average detection rate of 95.8% at a false positive rate of about 3.2% and quickly recovered from a temporary drift that compromised the performance of the individual models. The penalized weighted voting scheme with operation-lifetime worked efficiently in shifting power of decision to more updated classifiers and enabled AUC to recover from 0.89 in iter2 to 0.96 in iter4, and demonstrated that moderate updates at proper instances are sufficient to sustain robustness against non-stationarity of traffic. Comparative comparison also showed that the framework outperformed three state-of-the-art SDN IDS baselines by up to 14 percentage points in accuracy and 0.48 in F-score while maintaining low decision latency (≈50 ms), confirming that selective retraining with lifetime-aware voting is better in adaptability without the complete retraining expense. By updating only one or two ensemble members at each iteration, the system avoided approximately 25% of computational overhead while staying continuously available, and tracker-guided updates alleviated false alarms during drift without catastrophic forgetting. Synthetic traffic tests also verified strong generalization with an accuracy of 94.6% and an F-score of 0.946. Taken together, these findings constitute the contribution of this effort: a functional, scalable, and adaptable IDS architecture that strikes an effectiveness-efficiency-tradeoff between accuracy, efficiency, and robustness, which demonstrates that directed, performance-focused refreshes can render an SDN IDS agile in real-world settings. While the testing was limited to the InSDN dataset, broader validation to diverse topologies and adversarial settings must be the goal of future research, as well as optimization of retraining thresholds through meta-heuristic search and integration of explainable AI techniques to further improve interpretability for network managers. Briefly, the study provides compelling evidence that incremental learning with weighted ensemble voting is an acceptable path to resilient, low-cost intrusion detection in programmable networks that change by the hour.

Acknowledgment

I would like to express sincere gratitude to Dr. Fathy Eassa, Dr. Mohammed Y. Sahab, Dr. Abdullah Marsh, King Abdulaziz University, and the General Directorate of Passport, Ministry of Interior, Saudi Arabia, for their valuable support and contributions to this work.

References

[1] Z. Ahmad, A. Khan, C. 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, pp. 1–12, Jan. 2020.

[2] N. Ahmed, M. Ngadi, J. Sharif, S. Hussain, M. Uddin, M. Rathore, … and F. Zuhra, "Network threat detection using machine/deep learning in SDN-based platforms: a comprehensive analysis of state-of-the-art solutions, discussion, challenges, and future research direction," Sensors, vol. 22, no. 20, pp. 7896, Oct. 2022.

[3] M. Elsayed, N. Le-Khac, and A. Jurcut, "InSDN: a novel SDN intrusion dataset," IEEE Access, vol. 8, pp. 165263–165284, Sep. 2020.

[4] K. Wasielewska, D. Soukup, T. Čejka, and J. Camacho, "Evaluation of the limit of detection in network dataset quality assessment with PerQoDA," in Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases, pp. 170–185, Cham: Springer, Sept. 2022.

[5] M. Hadi and A. Mohammed, "A novel approach to network intrusion detection system using deep learning for SDN: futuristic approach," Comput. Sci. Inf. Technol., vol. 12, no. 11, pp. 1–6, Nov. 2022.

[6] H. Hassan, E. Hemdan, W. El-Shafai, M. Shokair, and F. El-Samie, "A survey on SDN-based intrusion detection systems on the Internet of Things: concepts, issues, and blockchain applications," Res. Square, pp. 1–20, Jul. 2021.

[7] A. Jose, L. Nair, and V. Paul, "Designing intrusion detection system in software defined networks using hybrid GWO-AE-RF model," Indian J. Comput. Sci. Eng., vol. 13, no. 6, pp. 1951–1966, Dec. 2022.

[8] M. Khairi, S. Ariffin, N. Latiff, K. Yusof, M. Hassan, F. AL-Dhief, … and M. Hamzah, "Detection and classification of conflict flows in SDN using machine learning algorithms," IEEE Access, vol. 9, pp. 76024–76037, Jun. 2021.

[9] G. Kumar and H. Alqahtani, "Machine learning techniques for intrusion detection systems in SDN: recent advances, challenges and future directions," Comput. Modeling Eng. Sci., vol. 134, no. 1, pp. 89–119, Jan. 2023.

[10] A. Mudgerikar, E. Bertino, J. Lobo, and D. Verma, "A security-constrained reinforcement learning framework for software defined networks," in Proc. IEEE ICC, pp. 1–6, Jun. 2021.

[11] A. Mudgerikar, E. Bertino, J. Lobo, and D. Verma, "A security-constrained reinforcement learning framework for software defined networks," in Proc. IEEE ICC, pp. 1–6, Jun. 2021.

[12] F. Noorbehbahani, A. Fanian, R. Mousavi, and H. Hasannejad, "An incremental intrusion detection system using a new semi-supervised stream classification method," Int. J. Commun. Syst., vol. 30, no. 4, pp. 1–15, Apr. 2015.

[13] A. Rayhan, S. Islam, S. Shatabda, A. Islam, and M. Robin, "Intrusion detection system in software-defined networks using machine learning and deep learning techniques: a comprehensive survey," TechRxiv Preprint, pp. 1–25, Sept. 2022.

[14] N. Shone, T. Ngoc, V. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Mar. 2018.

[15] T. Tang, L. Mhamdi, D. McLernon, S. Zaidi, M. Ghogho, and F. Moussa, "DeepIDS: deep learning approach for intrusion detection in software defined networking," Electronics, vol. 9, no. 9, pp. 1533, Sept. 2020.

[16] A. ALHILO, "Enhancing SDN anomaly detection: a hybrid deep learning model with SCA-TSO optimization," Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 5, pp. 1–10, May 2024.

[17] A. Abdulboriy, "An incremental majority voting approach for intrusion detection system based on machine learning," IEEE Access, vol. 12, pp. 18972–18986, Feb. 2024.

[18] A. Chetouane, "Risk-based intrusion detection system in software defined networking," Concurrency Comput. Pract. Exper., vol. 36, no. 9, pp. 1–15, May 2023.

[19] J. B. Christinal and A. A. Roseline, "Securing SDON with hybrid evolutionary intrusion detection system: an ensemble algorithm for feature selection and classification," Opt. Fiber Technol., vol. 93, pp. 104206, Jan. 2025.

[20] M. Elsayed, N. Le-Khac, M. Albahar, and A. Jurcut, "A novel hybrid model for intrusion detection systems in SDNs based on CNN and a new regularization technique," J. Netw. Comput. Appl., vol. 191, pp. 103160, Mar. 2021.

[21] H. Y. I. Khalid and N. B. I. Aldabagh, "A survey on the latest intrusion detection datasets for software defined networking environments," Eng. Technol. Appl. Sci. Res., vol. 14, no. 2, pp. 13190–13200, Apr. 2024.

[22] A. Mansoor, M. Anbar, A. A. Bahashwan, B. A. Alabsi, and S. D. A. Rihan, "Deep learning-based approach for detecting DDoS attack on

software-defined networking controller," Systems, vol. 11, no. 6, pp. 296, Jun. 2023.

[23] A. Almazyad, L. Halman, and A. Alsaeed, "Probe attack detection using an improved intrusion detection system," Comput., Mater. Continua, vol. 74, no. 3, pp. 4769–4784, Mar. 2023.

[24] Y. Xiao, X. Zhang, Z. Wang, and Y. Li, "[Title not available] (utilized InSDN dataset for feature selection and ML optimization in SDN IDS)," Unpublished, 2023.

[25] H.-M. Chuang, F. Liu, and C.-H. Tsai, "Early detection of abnormal attacks in software defined networking using machine learning approaches," Symmetry, vol. 14, no. 6, pp. 1178, Jun. 2022.

[26] O. Aouedi and K. Piamrat, "F-BIDS: federated blending based intrusion detection system for IoT/IIoT networks," Procedia Comput. Sci., vol. 213, pp. 275–284, Dec. 2023.

[27] H. Gâlmeanu and R. Andonie, "Concept drift adaptation with incremental–decremental SVM," Appl. Sci., vol. 11, no. 20, pp. 9644, Oct. 2021.

[28] H. Hassan, E. Hemdan, W. El-Shafai, M. Shokair, and F. El-Samie, "A survey on SDN-based intrusion detection systems on the Internet of Things: concepts, issues, and blockchain applications," Res. Square, pp. 1–20, Jul. 2021.

[29] S. A. Wadho, S. Ali, and A. A. A. Mohammed, "Secret sharing as a defense mechanism for ransomware in cloud storage systems," Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 10, pp. 1–10, Oct. 2024.

[30] R. Sebopelo, "An integrated framework for controllers placement and security in software-defined networks ecosystem," J. Inf. Syst. Informatics, vol. 6, no. 1, pp. 464–494, Feb. 2024.

[31] T. Wang, Q. Lv, B. Hu, and D. Sun, "A few-shot class-incremental learning approach for intrusion detection," in Proc. IEEE ICCCN, pp. 1–8, Aug. 2021.

[32] Malik, R. Q., et al. "A Novel Taneja Distance-based Classifier with PSO-Optimized Feature Selection for Efficient 5G Network Slicing." *International Journal of Intelligent Engineering & Systems* 18.6 (2025).

[33] Alsharfa, Raya Majid, et al. "Cellular-D2D resource allocation algorithm based on user fairness." *Electronics* 9.3 (2020): 386.

[34] Ali, Sijjad, et al. "CLDM-MMNNs: Cross-layer defense mechanisms through multi-modal neural networks fusion for end-to-end cybersecurity—Issues, challenges, and future directions." *Information Fusion* (2025): 103222.

[35] Alsharfa, Raya Majid. "Design and Performance of MC-CDMA Transceiver based LDWT–OFDM in Flat Fading Channel." *IOP Conference Series: Materials Science and Engineering*. Vol. 518. No. 5. IOP Publishing, 2019.

[36] Wadho, Shuaib Ahmed, et al. "Ransomware detection techniques using machine learning methods." *2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC)*. IEEE, 2024.

Comprehensive Analysis of YOLOv8 + DeepSORT for Vehicle Tracking: HOTA and CLEAR-Based Evaluation

I Nyoman Eddy Indrayana[1], Made Sudarma[2], I Ketut Gede Darma Putra[3], Anak Agung Kompiang Oka Sudana[4]
Engineering Science Department, Udayana University, Badung, Indonesia[1]
Electrical Engineering Department, Udayana University, Badung, Indonesia [2]
Information Technology Department, Udayana University, Badung, Indonesia[3, 4]

***Abstract*—This paper offers a thorough comparative investigation of the performance of a vehicle multi-object tracking system, incorporating various versions of the YOLOv8 detector (from 'n' to 'x') alongside the DeepSORT tracking algorithm. This study systematically assesses the impact of the trade-off between detector speed and accuracy on tracking metrics, utilising a real-world traffic video dataset from Bali. The assessment is performed utilising two fundamentally distinct metric frameworks: the traditional CLEAR metric (which includes MOTA) and the contemporary Higher Order Tracking Accuracy (HOTA) metric. The findings indicate that although the larger YOLOv8 model markedly enhances detection recall, particularly for smaller and more difficult items like motorcycles, tracking issues persist. The dual metric study provides significant insights: the HOTA measure demonstrates that car tracking has more associative stability (higher AssA scores) compared to motorbike tracking, which frequently experiences track fragmentation. In contrast, the detection-biased MOTA metric produces somewhat paradoxical outcomes, as motorbikes receive elevated scores due to enhanced detection accuracy (fewer false positives), therefore obscuring deficiencies in tracking consistency. This study concludes that HOTA offers a more comprehensive evaluation by differentiating between detection and association performance, so demonstrating that detection-only metrics like MOTA can yield an imperfect representation of actual tracking ability. These findings underscore the necessity of matching detector architecture and evaluation criteria with specific application requirements, particularly in safety-critical systems where identity consistency is essential.**

***Keywords*—Multi-object tracking; higher order tracking accuracy metric; CLEAR Metric; YOLOv8**

I. Introduction

Intelligent Transportation Systems (ITS) have emerged as a fundamental component in worldwide initiatives aimed at enhancing the efficiency, safety, and sustainability of transportation infrastructure [1]. Central to these systems is the capacity to dynamically observe, analyse, and regulate vehicle movements [1]. Multiple Object Tracking (MOT) has become a pivotal technology facilitating these sophisticated applications [1], [2]. The applications of MOT in Intelligent Transportation Systems are varied and significant, encompassing traffic flow analysis for congestion management, automated traffic enforcement, data collection for urban planning, and accident detection [3], [4]. The cornerstone of all these sophisticated systems is the accessibility of precise, real-time data regarding the movement of each vehicle on the road [1], [3].

The predominant and effective methodology in contemporary Multi-Object Tracking (MOT) is the tracking-by-detection paradigm [5]. This paradigm deconstructs the intricate tracking issue into two more feasible steps: initially, identifying all objects of interest (namely, cars) in each video frame; and subsequently, linking (associating) these detections across time to establish a coherent trajectory for each object [6]. The efficacy of this method is largely contingent upon the functionality of these two elements [7]. The integration of the YOLO (You Only Look Once) detector family with the SORT (Simple Online and Realtime Tracking) tracker family has gained significant popularity [8], showcasing the efficacy and efficiency of this paradigm across diverse real-world applications [8], [9].

Although several investigations have utilised the integration of YOLO and DeepSORT, most choose to employ a singular version of YOLO (e.g. YOLOv5) [10] and concentrate on enhancing the tracking algorithm itself. A comprehensive comparison review of the whole range of YOLOv8 devices, from the lightweight nano form to the robust extra-large variant, is still absent, particularly with on-road vehicle tracking.

Significantly, numerous current evaluations continue to depend extensively on traditional criteria like MOTA (Multiple Object Tracking Accuracy) [11], [12], [13], which are recognised for their substantial bias against detection performance. These measurements frequently conceal association failures (identification tracking) by aggregating them with detection mistakes [6]. The influence of detector scale selection on contemporary metrics like HOTA (Higher Order Tracking Accuracy), which distinctly differentiate between detection and association assessment, has not been extensively investigated [6].

This prompts a fundamental research inquiry: How does the compromise between detector precision and velocity (e.g. YOLOv8n versus YOLOv8x) affect overall tracking efficacy when assessed by a metric that equally prioritises detection accuracy and association consistency? The link between detection quality and tracking quality is hypothesised to be nonlinear. There may come a juncture where augmenting detector strength ceases to yield advantages or becomes

counterproductive, since more sensitive detectors can generate partial or ambiguous detections that obfuscate the tracker's association logic [6], [14].

This evaluation aims to identify specific trends in vehicle behaviour that may lead to accidents. Consequently, the findings of this research will enhance vehicle monitoring technology and facilitate the design of more efficient traffic safety systems. This research seeks to elucidate the correlation between vehicle trajectories and accident detection, as well as to formulate algorithms that are more responsive to fluctuating road circumstances.

This study significantly contributes to the fields of computer vision and intelligent transportation systems through the following aspects:

- Comprehensive Empirical Evaluation: Provides an in-depth empirical evaluation of five YOLOv8–DeepSORT combinations on a challenging and contextually pertinent real-world vehicle monitoring dataset gathered from Bali, Indonesia.
- Dual Metric Analysis: Performs a thorough performance evaluation utilising both the CLEAR metric and the more extensive HOTA metric, offering a detailed perspective on tracker behaviour that is unattainable by a singular metric alone.

The subsequent sections of this work are structured as follows. Section II comprises a literature review that elucidates pertinent studies, emphasizing advancements in object identification and tracking, specifically with YOLO and DeepSORT. Section III delineates the proposed approach and the materials utilized, encompassing dataset preparation, model configuration, and evaluation protocol. Section IV presents the experimental data and offers a comparative analysis utilizing the HOTA and CLEAR metrics. Section V presents the conclusions of this article and delineates prospective avenues for further research.

II. Literature Review

A. Advancement of Object Detection Utilising YOLO

Object detection is a crucial component in numerous computer vision applications, encompassing object tracking. The YOLO (You Only Look Once) architecture, introduced in 2016 with YOLOv1, represented a substantial advancement in real-time object recognition with an efficient one-stage detection methodology [15]. The progression of YOLO persists with enhancements in design and training methodologies in YOLOv2 [16], YOLOv3 [17], YOLOv4 [18], YOLOv5 [19], YOLOv6 [20], and YOLOv7 [21]. Ultimately, YOLOv8 provides a flexible framework featuring an extensive array of models tailored for diverse computational and accuracy needs [22], [23], [24].

The progression of YOLO (You Only Look Once) from version 1 to 8 signifies a swift enhancement in real-time object identification. YOLOv1 initiated a paradigm shift by conceptualising detection as a singular regression task, directly forecasting bounding boxes and classes from the entire image in one iteration, hence achieving much greater speed compared to earlier two-stage detectors. Subsequent iterations, including YOLOv2 through YOLOv5, progressively enhanced this architecture by incorporating anchor boxes, more resilient backbones like CSPDarknet53 [25], and feature aggregation networks such as PANet to optimise the balance between accuracy and speed [26]. YOLOv8 achieved notable advancements by enhancing the backbone and neck for improved feature extraction and implementing a pivotal modification through the adoption of an anchor-free detection head [25]. This anchor-free methodology streamlines the prediction process and enhances the model's flexibility to objects of diverse scales and aspect ratios, distinguishing it from its predecessors that depended on specified anchor boxes.

Recent advancements indicate that augmenting YOLOv8 with supplementary modules can enhance performance in intricate circumstances. Abdullah N. Alhawsawi et al. introduced an Enhanced YOLOv8 using a Context Enrichment Module (CEM) to enhance crowd counting in drone imagery by more effectively differentiating small, dense targets[27]. This method enhances detection precision, while our research combines YOLOv8 with DeepSORT to maintain identity coherence in vehicle tracking. These studies demonstrate two alternative approaches: enhancing detector capability or integrating detection with robust tracking for dependable multi-object surveillance in intelligent transportation systems.

B. Intersection over Union (IoU)

In contemporary object detection systems, a crucial element that quantifies spatial accuracy between predictions and ground truth annotations is Intersection over Union (IoU). Intersection over Union (IoU) is a crucial assessment metric in numerous deep learning-based object detection algorithms [28], notably the YOLO (You Only Look Once) model family and its most recent iteration, YOLOv8. In the realm of vehicle recognition within roadway settings, pertinent to Intelligent Transportation Systems (ITS) applications or traffic surveillance, elevated spatial precision is essential to enable the system to accurately identify and detect cars across diverse environmental circumstances.

In the evaluation of object detection performance, Intersection over Union (IoU) is commonly employed as a metric to quantify the overlap between predicted and actual bounding boxes. The Intersection over Union (IoU) is determined by the ratio of the intersection area of the predicted and ground truth boxes to their total area. A high IoU value signifies a more precise prediction in object localisation. The Intersection over Union (IoU) is frequently employed as a criterion to ascertain whether a detection is classified as positive or negative in the computation of accuracy and recall metrics. The accuracy and precision of object detection, often assessed using IoU-based metrics, directly influence the efficacy of subsequent tracking tasks. Inaccurate detection may result in track fragmentation and misidentification during the tracking process. Intersection over Union (IoU) is the ratio of the overlapping area of two bounding boxes, specifically the model prediction and the ground truth annotation, to the total area of their union. The mathematical formulation of IoU is as follows:

$$IoU = \frac{|A_p \cap A_{gt}|}{|A_p \cup A_{gt}|} \quad (1)$$

The bounding box of the model prediction results is denoted as A_p, while the bounding box of the ground truth data is represented as A_{gt}, The symbol $\cap$ signifies the overlapping area (intersection), and $\cup$ denotes the total combined area of the two boxes (union).

In computational implementation, if each bounding box is denoted by its border coordinates $(x_{min}, y_{min}, x_{max}, y_{max})$, the intersection area is computed as (2):

$$Area_{inter} = max\left(0, min\left(x_p^{max}, x_{gt}^{max}\right) - max\left(x_p^{max} - x_{gt}^{max}\right)\right) x\ max\left(0, min\left(y_p^{max}, y_{gt}^{max}\right) - max\left(y_p^{max} - y_{gt}^{max}\right)\right) \quad (2)$$

and the union area is computed as (3):

$$Area_{union} = Area_p + Area_{gt} - Area_{inter} \quad (3)$$

The IoU value spans from 0 to 1, with 1 signifying complete overlap and 0 denoting no overlap at all.

C. Object Tracking with DeepSORT

The SORT (Simple Online and Realtime Tracking) algorithm, developed by Bewley in 2016, provides an efficient solution for the multiple object tracking (MOT) task by employing object detection and a Kalman filter for movement prediction, alongside the Hungarian algorithm for associating detections across successive frames [8]. Nevertheless, SORT is significantly reliant on detection quality and is less proficient in managing variations in object appearance and occlusion. DeepSORT addresses this restriction by including appearance cues derived from a deep neural network designed for re-identification tasks. The utilisation of these visual attributes enables DeepSORT to enhance object tracking, even amongst alterations in appearance or transient occlusion [29].

DeepSORT is constructed upon the SORT framework, which utilises bounding box location estimation via a Kalman filter and employs the Hungarian method for data association [10], [11]. The primary innovation of DeepSORT is the utilisation of appearance information derived from objects through a Convolutional Neural Network (CNN) [30]. These visual attributes offer supplementary information during the association process, especially in preventing misidentification when two items overlap or momentarily vanish from the frame.

The tracking procedure utilising DeepSORT comprises three primary components. Initially, object identification is executed utilising detection architectures like YOLOv8, SSD, or Faster R-CNN. Subsequently, the item's position in the subsequent frame is forecasted employing a Kalman Filter. Third, an association process is conducted between the new detection and the existing track, using spatial metrics and visual similarities. This enables DeepSORT to preserve object identity with greater precision and demonstrates resilience to visual noise.

The Kalman Filter is essential for forecasting the position of the monitored item between successive video frames. This method is executed iteratively. This filter determines the object's position through the coordinates of the bounding box's centre (u, v), the aspect ratio (width divided by height is γ), the height of the bounding box (h) and the predicted rate of change for each of these components $(\dot{u}, \dot{v}, \dot{\gamma}, \dot{h})$. Utilising this predictive capability, DeepSORT can adeptly forecast the object's movement within the video. The state representation formula is articulated in the subsequent (4):

$$x = \left[u, v, \gamma, h, \dot{u}, \dot{v}, \dot{\gamma}, \dot{h}\right]^T \quad (4)$$

The state vector x mathematically represents the state of a monitored object at a certain moment within the DeepSORT Kalman Filter. This vector comprises eight elements, each signifying the following:

u : The central horizontal coordinate (centre x-coordinate) of the object's enclosing box. This denotes the object's horizontal placement within the video frame.

v : The central vertical coordinate (central y-coordinate) of the object's enclosing box. This denotes the object's vertical placement within the video frame.

γ : The aspect ratio of the bounding box, determined by dividing its width by its height. This conveys details regarding the object's proportional configuration.

h : The elevation of the object's bounding box. This conveys details regarding the object's vertical dimension.

$\dot{u}$: The speed of the central horizontal coordinate. This assesses the velocity and direction of the horizontal movement of the bounding box's centre.

$\dot{v}$: The speed of the central vertical coordinate. This assesses the velocity and direction of the vertical movement of the bounding box's centre.

$\dot{\gamma}$: The rate of variation in the aspect ratio. This assesses the temporal alterations in the object's proportionate form.

$\dot{h}$: The rate of alteration in the elevation of the bounding box. This assesses the temporal variation in the object's vertical dimensions..

^{T} : The notation at the vector's conclusion signifies that it is a transposed vector, represented in column format.

Appearance features are derived by extracting patches from the identified bounding box and processing them through a CNN architecture, such as ResNet-50 or MobileNet. A CNN produces a fixed-dimensional embedding vector, usually 128 dimensions that encapsulates the visual attributes of the object. Each track maintains a record of these embeddings, utilised to assess appearance similarity among frames.

In the DeepSORT algorithm, when numerous objects are detected in a video frame and we aim to track their identities across time, it is necessary to determine which object detection in the current frame corresponds to which object track from the preceding frame. The process of data matching, or association, is essential to prevent the misidentification of objects. DeepSORT employs two essential measures to facilitate precise association determinations: mahalanobis distance and cosine similarity [10].

Mahalanobis distance quantifies the proximity of the current object detection to the predicted position of the tracked object. In contrast to the conventional Euclidean distance, Mahalanobis distance incorporates uncertainty in the projected object position [31]. The covariance matrix obtained from the Kalman Filter employed in tracking signifies this uncertainty. Conversely, when the tracker exhibits high confidence in its forecast (shown by minimal covariance), a minor positional discrepancy will yield a substantial Mahalanobis distance. Conversely, if the tracker exhibits diminished confidence (high covariance), a greater positional discrepancy may still be seen as proximate. The Mahalanobis distance formula (5) between detection d_j, prediction y_i , utilising the covariance matrix S_i can be expressed as [32]:

$$d^{(1)}(i,j) = \left(d_j - y_i\right)^T S_i^{-1} \left(d_j - y_i\right) \qquad (5)$$

DeepSORT utilises both location and appearance information from each object. Each identified object has its visual characteristics retrieved via a deep neural network. These features are vectors that denote the object's distinctive visual attributes. Cosine Similarity is employed to assess the similarity of appearance features between the current detection and the appearance features of the object's trajectory recorded from prior frames. Cosine Similarity quantifies the angle between two feature vectors. When the angle is minimal (cosine value approaching 1), the two visual characteristics exhibit significant similarity, suggesting they are likely the same object. When the angle is substantial (cosine value approaching -1), the visual characteristics are markedly distinct.

The amalgamation of these two metrics generates an association cost matrix, which is resolved with the Hungarian Algorithm. To evaluate the visual resemblance between the bounding box of the currently tracked object (i) and the bounding box of the newly detected object (j), we employ an equation $d^{(2)}(i,j)$ (6) that computes the cosine distance, which is the inverse of the cosine similarity, between their appearance feature vectors $(r_j \; and \; r_i)$. The dot product $(r_j^T r_i)$ yields the cosine similarity value, and by subtracting it from 1, we derive the cosine distance. A small distance signifies maximal similarity in visual characteristics.

$$d^{(2)}(i,j) = min\left\{1 - r_j^T r_k^{(i)} \middle| r_k^{(i)} \epsilon R_i\right\} \qquad (6)$$

Upon establishing the association, the system classifies the track as matched, mismatched, or a novel detection. Upon a detection aligning with an existing track, the Kalman filter is revised, and the appearance features are recorded. In the absence of corresponding detections, the track persists for some frames prior to being eliminated. Conversely, novel, unassociated detections will generate new tracks if their confidence level is sufficiently high. Parameters, including maximum age, minimum hits, and matching threshold significantly influence the system's susceptibility to noise and visual disruptions.

D. Higher Order Tracking Accuracy (HOTA) Metric

Higher Order Tracking Accuracy (HOTA) is a principal evaluation statistic intended to furnish a singular, equitable assessment of numerous object tracking efficacy [6]. HOTA explicitly integrates accurate detection, proper association between detection and tracking, and exact localisation into a singular metric. A high HOTA is intuitively attained when the tracking system effectively identifies a significant percentage of target objects, persistently preserves their identities across frames, and accurately anticipates bounding boxes. HOTA is determined by the square root of the product of Detection Accuracy (DetA) and Association Accuracy (AssA), imposing a penalty for subpar performance in either dimension.

Detection Accuracy (DetA) is a sub-metric of HOTA that particularly evaluates the efficacy of the tracking system in identifying the proper item in each frame. DetA is computed by comparing the quantity of accurate detections (True Positives) to the total count of ground truth objects and resultant detections (True Positives + False Negatives + False Positives). In essence, DetA resembles the F1 score for detection tasks, balancing detection precision (minimising False Positives) and detection recall (minimising False Negatives). A high DetA signifies that the system can accurately identify the majority of target objects while minimising false detections.

Association Accuracy (AssA) is a sub-metric of HOTA that evaluates the precision with which a tracking system preserves the right object to identify throughout video frames. AssA evaluates the quality of the correlation between a detection in the current frame and the established track from the preceding frame. Calculating AssA entails contrasting the quantity of accurate associations with the entire number of potential linkages. An accurate association transpires when a detection is linked to the appropriate ground truth track. A high AssA signifies that the system can constantly monitor objects without frequent identity transitions or track fragmentation.

Localisation Accuracy (LocA) is a sub-metric of HOTA that assesses the precision with which the tracking system's bounding boxes identify the target item. LocA is generally computed as the mean Intersection over Union (IoU) between the predicted bounding box and the ground truth bounding box for all accurate detections. A high LocA signifies that the system not only accurately identifies items but also precisely forecasts their locations and dimensions.

False Positives (FP) refer to the detections produced by the tracking system that do not align with any actual ground truth objects. FP denotes the existence of an object that is, in reality, nonexistent (according to the ground truth annotation). A lower FP number indicates greater detection precision of the system.

False Negatives (FN) refer to the quantity of actual items that the tracking system failed to detect. FN denotes the system's inability to identify an object that was genuinely present in the frame. A lower FN value corresponds to an increased detection recall of the system.

Identity Precision (IDP) is a parameter that assesses the purity of the tracks produced by the tracking system. IDP is determined by the ratio of accurate associations to the total number of detections made by the system. A high IDP indicates that the majority of the generated tracks accurately reflect the genuine identify of the object.

Identity Recall (IDR) is a metric that evaluates the efficacy of a tracking system in preserving accurate ground truth identities. IDR is determined by the ratio of correct associations

to the total count of ground truth items. A high IDR signifies that the system effectively monitors the majority of ground truth objects while preserving their identities.

Identity Switches (IDS) refer to the frequency with which a monitored object's identity is erroneously altered throughout a video sequence. A low IDS signifies enhanced stability and consistency in the performance of association for preserving object identities.

E. CLEAR Metric

The CLEAR (Classification of Events, Activities, and Relationships) metric serves as a benchmark for the quantitative and objective assessment of moving object tracker performance [33]. Multi-Object Tracking Accuracy (MOTA) serves as the principal composite statistic inside the CLEAR evaluation framework [34]. MOTA comprises three principal types of mistakes in tracking: false positives (FP), false negatives (FN), and identity switches (IDS) [33]. MOTA is formally computed as (7):

$$MOTA = 1 - \frac{|FN|+|FP|+|IDSW|}{gtDet} \quad (7)$$

In frame t, FN, FP, and IDSW denote the quantities of false negatives, false positives, and identity switches, respectively, whereas gt represents the count of ground truth objects in frame t. MOTA delivers a singular metric that evaluates both detection precision and identification coherence across time. An elevated MOTA value signifies superior tracking performance.

Multi-Object Tracking Precision (MOTP) assesses the accuracy of successfully localised tracked objects, excluding errors in detection or identity [35]. MOTP is determined by the average overlap, typically employing Intersection over Union (IoU), between the predicted bounding boxes and the ground truth bounding boxes for all true positives over all frames. MOTP can be formally expressed as follows (8) [5]:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \quad (8)$$

where, $d_{t,i}$ is the distance (e.g. 1−IoU) between detection i and the corresponding ground truth at frame t, and C_t is the number of correct matches at frame t. MOTP indicates the precision of the system in localising successfully tracked objects, with elevated values signifying superior localisation.

III. Method and Material

A. MOT Step

The implementation of a Multiple Object Tracking (MOT) system for road vehicles commences with the acquisition of an input video stream, sourced either directly from a camera or from pre-existing footage. The video is processed by decomposing it into a sequence of frame acquisitions, each of which is independently and sequentially processed. The last essential phase is object detection, which in this research was performed utilising different forms of the YOLOv8 architecture (nano, small, medium, large, or extra-large). The selection of the YOLOv8 variation considerably influences the equilibrium between inference speed and detection accuracy, which is the primary focus of this evaluation. The YOLOv8 detection results comprise detection output, a collection of bounding boxes around each identified vehicle, confidence ratings reflecting the amount of detection certainty, and class labels categorising the vehicle type.

Following detection, feature extraction is conducted for each identified object utilising a deep neural network to obtain appearance features, a crucial component of the DeepSORT algorithm. This pre-trained deep neural network produces Appearance Features, which are vector descriptors that distinctly characterise the visual attributes of each vehicle. Simultaneously, State Estimation and Prediction are executed for each existing object track utilising a Kalman Filter. This filter assesses the present object state (position, velocity, etc.) and forecasts the anticipated object states in the subsequent frame.

The subsequent phase is Association, which seeks to link the object detection in the present frame with the track prediction from the preceding frame. This procedure utilises the Hungarian Algorithm, which operates on a cost matrix. This cost matrix evaluates two primary metrics: distance and appearance are assessed based on the disparity in visual characteristics between the detection and the track, while the Mahalanobis Distance (Motion) quantifies the deviation between the detected position and the motion prediction, incorporating the uncertainty inherent in the Kalman filter prediction [36]. The Hungarian algorithm subsequently seeks the assignment with the lowest cost. Successfully linked detections are utilised for Track Updates, wherein the Kalman Filter of the track is revised using the most recent detection data, enhancing the estimation of the object's state. Ultimately, Track Management oversees the lifecycle of the object track through Initialisation for consistent new detections and Termination for tracks that remain undetected across multiple frames. The ultimate outcome of this method is Output: Tracked Objects, which displays the movement trajectory of each identified vehicle together with unique IDs that are preserved throughout the film. The phases of object tracking utilising Yolov8 and DeepSORT are illustrated in Fig. 1.

B. YOLOv8 Object Detection

The object detection procedure utilising YOLOv8 on a video stream commences with the acquisition of an input video stream, which may originate from a live roadside security camera or a recorded video. The video stream is segmented into a sequence of frame acquisitions, with each frame serving as a static visual analysis unit that will be processed sequentially by the YOLOv8 model. The subsequent stage is grid division, wherein each frame is partitioned into an S×S grid of cells [37]. This division seeks to localise items into designated regions of the frame, with each grid cell tasked with anticipating the object's centre contained within it.

YOLOv8 employs an anchor-free prediction methodology, distinguishing itself from its predecessors by explicitly forecasting the centre, height, and breadth of the bounding box in relation to each grid cell. This method offers enhanced adaptability in managing discrepancies in object dimensions and proportions. Each input frame undergoes a convolutional feature extraction process utilising the YOLOv8 CNN architecture [38]. This network systematically extracts visual information from low-level elements (edges and corners) to high-level

components (object-specific shapes). YOLOv8 utilises an efficient backbone architecture, such as CSPDarknet [39] or the C2f module to achieve a balance between speed and accuracy.

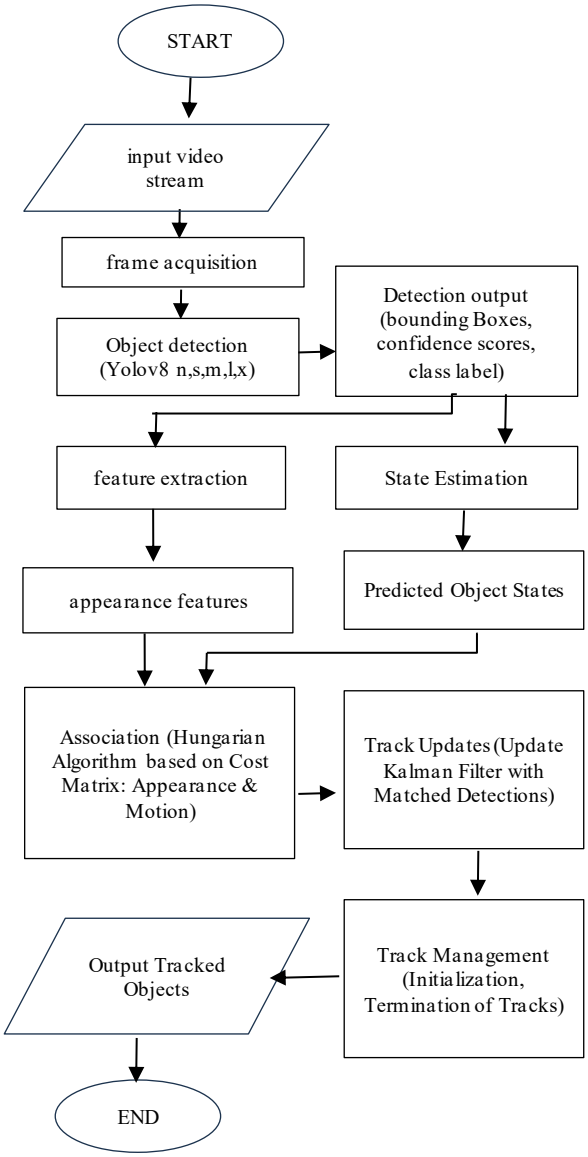


Fig. 1. Multi object tracker step.

YOLOv8 concurrently produces a Prediction per Grid Cell for each grid cell, encompassing the coordinates (x,y,w,h) of the prospective bounding box, an Objectness Score reflecting the confidence level of the object within the bounding box, and Class Probabilities that denote the conditional probability for each object class upon detection. Due to the possibility of numerous detections of the same object from several grid cells or overlapping bounding box predictions, Non-Maximum Suppression (NMS) is utilised. This procedure iteratively identifies the bounding box with the highest confidence score and discards other bounding boxes that exhibit substantial overlap (measured by Intersection over Union - IoU) and possess lower confidence scores. This procedure persists until solely the most precise bounding box for each identified object is preserved. The culmination of this phase is the Detection Output, comprising a collection of filtered bounding boxes, each associated with a high confidence score and accompanying vehicle class label, which will subsequently function as input for the DeepSORT tracking phase.

C. Tracking Evaluation

Assessing the efficacy of a Multiple Object Tracking (MOT) system is essential for comprehending the proficiency of your algorithm in tracking things within video [5]. This process involves comparing the MOT system's output with manually annotated ground truth data. Metrics such as HOTA and CLEAR are employed to assess several dimensions of tracking precision.

The evaluation phase commences with Ground Truth Data (Manual Annotations). This constitutes the basis of the comprehensive assessment, featuring detailed manual annotations for each pertinent object (e.g. motorbikes, vehicles, buses, trucks) in every video frame. These annotations comprise precise bounding boxes and constant unique identifiers for each object throughout the video frames. Upon completion of the annotations, the MOT data must be prepared to ensure compatibility with the evaluation code. A standard format often comprises the frame number, object ID, and bounding box coordinates (x, y, width, height).

Subsequently, we present the Tracking Result Data (MOT System Output). This is the output produced by your MOT system, namely a synthesis of YOLOv8 and DeepSORT. This data includes frame-by-frame details regarding the identified and monitored objects, together with bounding boxes and IDs allocated by the tracking algorithm. Similar to the GT data, this tracking data must be prepared to adhere to the HOTA and CLEAR assessment requirements.

Upon formatting both data sets, the GT data and the tracking data, we proceed to the HOTA Evaluation phase. The HOTA code implementation will juxtapose the tracks produced by your system with the ground truth tracks for each object across the video sequence. The outcome of this assessment is HOTA Metrics, an extensive array of metrics that offers a thorough analysis of tracking performance. The metrics encompass: HOTA (which balances detection and association accuracy), DetA (detection accuracy), AssA (association accuracy), LocA (bounding box localisation accuracy), FP (false positives), FN (false negatives), IDP (identity precision), IDR (identity recall), and IDS (identity switches) [40], [41].

The CLEAR Metrics review is conducted concurrently with or subsequent to the HOTA review. GT data and tracking outcomes are used to compute these traditional metrics. The resulting CLEAR metrics encompass: MOTA (Multi-Object Tracking Accuracy), which integrates detection and association errors; MOTP (Multi-Object Tracking Precision), which assesses bounding box accuracy; MT (Mostly Tracked), the proportion of ground truth objects that are predominantly tracked; PT (Partially Tracked), the proportion of ground truth objects that are partially tracked; ML (Mostly Lost), the proportion of ground truth objects that are predominantly untracked; FP and FN (as in HOTA); IDS (number of identity switches); and Frag (Fragmentation), the frequency with which a track is interrupted and subsequently resumed.

The concluding phases involve the analysis and interpretation of HOTA results and CLEAR results. The metric data from both sets are assessed to comprehend the overall efficacy of the tracking system. The emphasis for HOTA is on the aggregate HOTA value, including the contributions of DetA and AssA. Analysing HOTA values across many YOLOv8 versions will identify the variation that provides optimal tracking performance. Alternative measurements offer supplementary insights into particular facets. Likewise, CLEAR metrics, particularly MOTA, offer a comprehensive summary, whereas MOTP emphasises bounding box precision. MT, PT, and ML measurements offer insights into the track quality of individual objects. The Mostly Tracked (MT) statistic quantifies the proportion of ground truth objects that are effectively monitored for the majority of their lifespan. An object is deemed "mostly tracked" if it is effectively linked to a track for a minimum of 80% of its overall duration in the video. MT offers insight into the system's capacity to sustain long-term monitoring of a singular item. Conversely, Mostly Lost (ML) quantifies the proportion of ground truth objects that are not monitored for the majority of their lifespan [31]. ML denotes the frequency with which the system fails to sustain long-term tracking of an item. Partially Tracked (PT) denotes the proportion of ground truth objects that do not belong to either the MT or ML categories. These things are monitored for a substantial duration of their existence, yet insufficiently to be categorised as MT or marginally enough to be designated as ML

False Positives (FP) refer to the aggregate number of detections produced by the tracking system that do not align with any actual objects in the complete video sequence. FP denotes erroneous detections, or backdrops erroneously identified as objects. False Negatives (FN) refer to the total count of actual objects that the tracking system fails to detect during the whole video sequence. FN denotes the system's inability to identify an object that is genuinely present. Identity Switches (IDS) refer to the cumulative instances in which a monitored object's identity is erroneously altered over the video sequence. A low IDS signifies enhanced identity consistency and the system's capacity to preserve individual object trajectories without interchanging them with those of other objects. Fragmentation (Frag) quantifies the frequency with which an object's trajectory is disrupted and subsequently re-established. Fragmentation transpires when an object is monitored, thereafter vanishes (lacking association with a detection in the subsequent frame), and later reemerges with the identical ID. Minimal fragmentation signifies a more stable and uninterrupted trajectory. A conclusion regarding Tracking Performance Comparison is reached through this analysis, pinpointing which YOLOv8 variants consistently yield the highest HOTA values and other pertinent metrics for tracking motorcycles, cars, buses, and trucks on highways, while also addressing the trade-off between accuracy and speed. The illustration depicting the phases of the object tracking evaluation procedure is presented in Fig. 2.

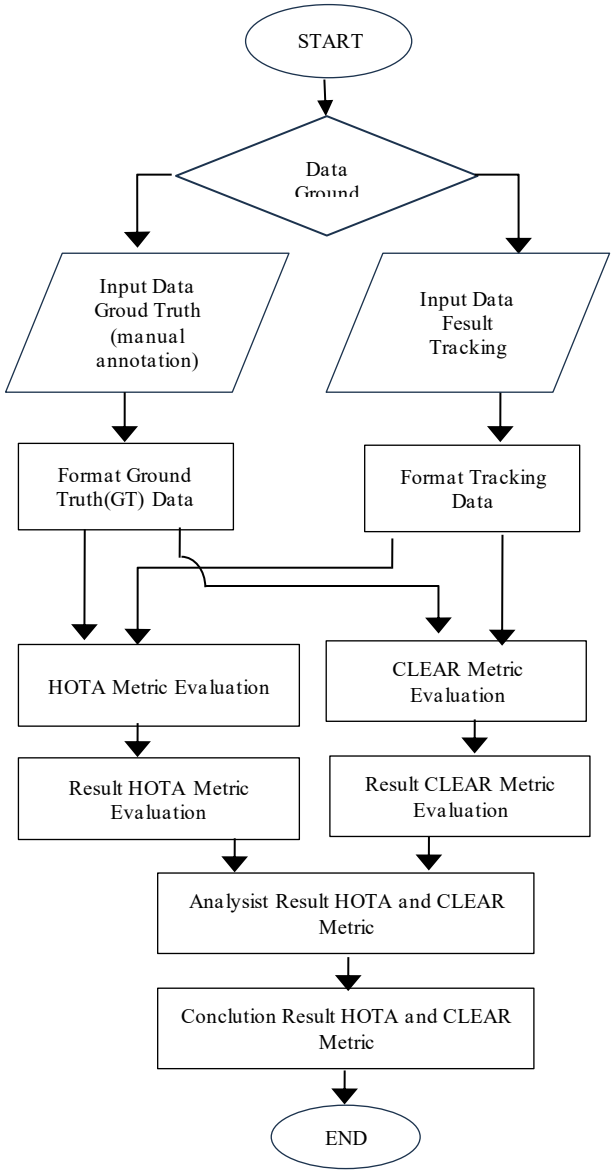


Fig. 2. Object tracking evaluation process.

D. Material Research

This work employs a relevant and extensive video dataset for the analysis of vehicle object tracking. The video data included in this experiment was specifically obtained from CCTV security cameras operated by the Bali Provincial Transportation Agency (https://balisatudata.baliprov.go.id/ peta-cctv). This study employs a collection of 980 real-world traffic surveillance photos, featuring 8,829 identified items categorized into four classes: cars, motorcyclists, buses, and trucks. To ensure model robustness, data were collected from two types of road environments: straight roads and intersections, thereby capturing diverse vehicle dynamics and spatial configurations.

The dataset encompasses differences in traffic flow conditions—spanning free-flow, moderate, and congested traffic—allowing for the evaluation of the tracking model under different density levels. The video data collection period took place in September 2024. This period was chosen to represent typical traffic patterns in the region, omitting substantial fluctuations that may occur during vacations or special events. It is important to acknowledge that all data were collected under daytime conditions, so presenting nocturnal scenarios as a forthcoming issue for future research.

This video dataset predominantly comprises two primary categories: motorbikes and cars, which are the subject of this research. Bali possesses a substantial motorbike demographic that often engages with bigger four-wheeled vehicles, including cars, buses, and trucks. The varied dimensions, forms, and motion trajectories of these two vehicle categories pose intriguing problems for object tracking algorithms. This dataset was created to assess the efficacy of tracking algorithms in differentiating and maintaining the identity of diverse vehicle types in congested traffic conditions.

The gathered video data is subsequently utilised as input for the object tracking system created in this study, which incorporates the YOLOv8 object recognition framework and the DeepSORT tracking algorithm. Each video is segmented into a sequence of individual frames, which are subsequently analysed by the YOLOv8 model to identify the presence and location of cars. The YOLOv8 detector's output, comprising bounding boxes, confidence scores, and class labels for each identified vehicle, is subsequently fed into the DeepSORT algorithm. DeepSORT uses this information, in conjunction with visual appearance attributes derived from each detection and motion prediction via a Kalman filter, to temporally track objects while preserving the distinct identification of each vehicle across video frames.

The utilisation of authentic video footage from a congested traffic setting in Bali enhances the authenticity and practical significance of this study's findings. The variety of traffic circumstances and vehicle types in the dataset enables thorough evaluation of the tracking algorithm's performance in practical situations. Moreover, the use of data from public transportation infrastructure, specifically CCTV from the Department of Transportation, underscores the prospective implementation of this object tracking technology in forthcoming traffic management and transportation monitoring systems. The dataset's quality and attributes establish a crucial basis for assessing performance and recognising potential enhancements in the suggested object tracking methodology.

Moreover, the acquisition and utilisation of this video data were executed with appropriate consideration for pertinent ethical and privacy issues. The data utilised is publicly available and sourced from surveillance cameras deployed for traffic monitoring. This research does not entail the acquisition of personal data or information that could identify specific persons beyond the context of their automobiles on the roadway. The research primarily concentrates on the analysis and tracking of various vehicle types for scientific and technological advancement. This dataset is deemed representative and sufficient for assessing the efficacy of object tracking algorithms within the framework of intense and varied urban traffic.

IV. Result

A series of tests performed using 30-second video footage at a Bali crossroads yields significant visual insights into the evolutionary performance of several YOLOv8 model variants integrated with the DeepSORT algorithm. The quantitative assessment exclusively targets cars and motorbikes, the predominant vehicle categories at the site, while the visualisation additionally illustrates the system's efforts to identify other classes, including buses and trucks. Each graphic, depicting progressively intricate model variants (from n to x), clearly demonstrates the trade-off among speed, detection accuracy, and classification reliability, closely correlating with the data shown in the HOTA and CLEAR metrics tables. We executed the tests in succession, first with Yolov8 version n and progressing to version x. The identifying number (ID) of each object in every test was randomly and uniquely created. Subsequently, we juxtaposed the visualisation outcomes within the identical frame.

To assess the model's detection efficacy on our unique dataset, we computed the mean Average Precision metric at a threshold of 0.50 (mAP50) for all four specified vehicle categories. This metric quantitatively assesses the model's accuracy in object localisation, with a detection deemed correct if the overlap between the predicted box and the ground-truth box is above 50%. The visual outcomes of this accuracy evaluation for each category are elaborated in Fig. 3.

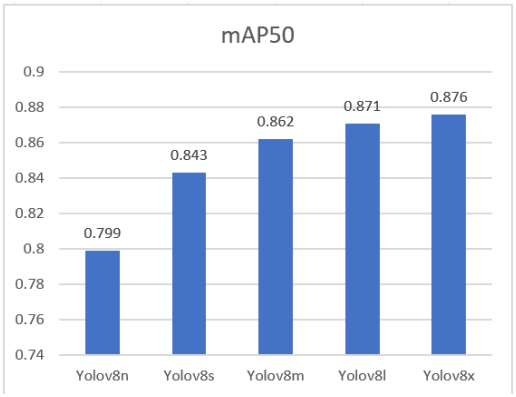


Fig. 3. mAP50 of Yolov8 variant.

Fig. 4 illustrates the outcomes of the YOLOv8n + DeepSORT model, highlighting the efficacy of the most lightweight and rapid variant. As a baseline model, its performance demonstrates notable shortcomings. The most significant inaccuracy was a false positive identification of object ID 415, erroneously classified as a "bus." In reality, this object was a massive billboard featuring a complex graphic, suggesting that the `n` model had difficulty differentiating between substantial vehicles and misleading background elements. Additionally, there were multiple false negatives, with roughly four motorcycles completely undetected. The inability to identify smaller, rapidly moving objects such as motorcycles is a prevalent deficiency in lighter models, which immediately resulted in diminished Detection Recall (DetRe) and MOTA scores in the quantitative assessment.



Fig. 4. Result from YOLOv8n + DeepSORT.

Fig. 5, which employs the YOLOv8s + DeepSORT model, shows that a progressive enhancement is observed. The count of undiscovered motorcycles decreases to three, signifying that the marginally more intricate 's' model had superior detection skills, hence diminishing the incidence of false negatives. Nonetheless, classification issues persist. Object ID 343, a white van, is erroneously categorised as a "bus". This categorisation of vehicles with unclear shapes (between a huge automobile and a small bus) is a persistent issue in the tests. Although these errors, when identified, do not directly influence detection metrics such as "DetA", they may affect evaluation if conducted purely on a per-class basis.

Fig. 6 presents the outcomes derived from the YOLOv8m combined with the DeepSORT model. There is a persistent misclassification, as the identical white van (now identified as 349) is once more erroneously categorised as a "bus". Nevertheless, it is evident that all motorcycles in the primary lane have been accurately detected, signifying an enhancement in detection recall relative to the 'n' and 's' variants. The enhancement in the capacity to identify all pertinent target objects is substantial and is evidenced by the increase in the "DetA" and "MOTA" scores in the metrics table. This suggests that with an increase in model size, its capacity to manage smaller and partially obscured objects becomes more resilient.

The trend of enhanced detection performance is notably illustrated in Fig. 7, produced by the YOLOv8l + DeepSORT model. An important discovery is that all vehicles, including motorcyclists, are accurately recognised and allocated bounding boxes. This is a notable accomplishment, indicating that the 'l' model possesses adequate representational capacity to tackle the detection issues at this congested intersection, successfully eradicating false negatives for this frame. Classification issues remain, as the white van (ID 290) is now erroneously categorised as a "truck". This underscores that enhancements in detection capabilities do not necessarily result in proportional advancements in classification accuracy for ambiguous instances.

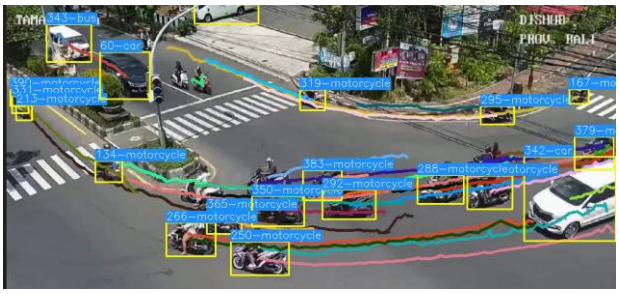


Fig. 5. Result from YOLOv8s + DeepSORT.

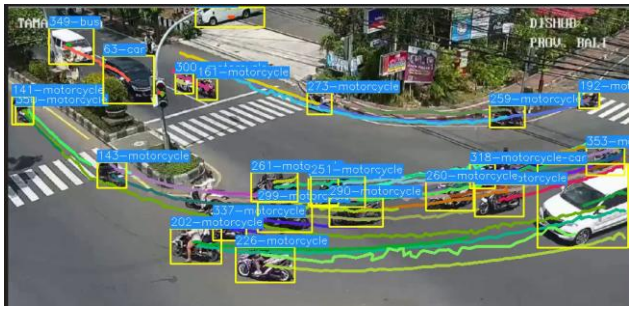


Fig. 6. Results from YOLOv8m + DeepSORT.

Ultimately, Fig. 8 illustrates the performance of the YOLOv8x + DeepSORT model, the most formidable option. Similar to the 'l' model, the 'x' model effectively identified all automobiles within the frame, demonstrating that the larger model markedly excels in recall performance. Nonetheless, it erroneously categorised a white van (ID 284) as a "truck". This indicates that, despite the model's extensive capacity, differentiating between visually analogous vehicle subclasses (such as vans, minibuses, and light trucks) continues to pose a significant challenge and may necessitate more varied training data or targeted fine-tuning methodologies. The visual progression from Fig. 4 to Fig. 8 clearly illustrates that augmenting the complexity of the YOLOv8 model enhances recall detection skills; yet, persistent challenges with ambiguous object classification underscore the necessity for future enhancement.

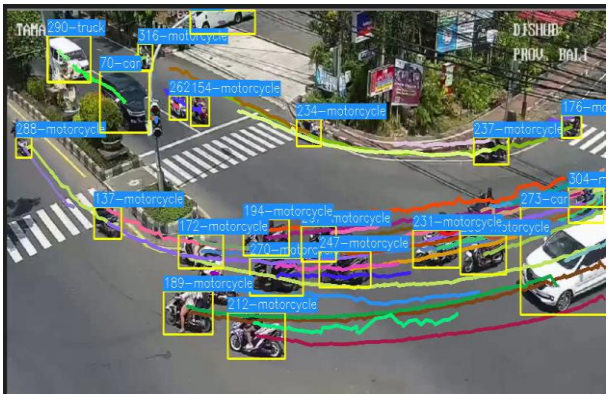


Fig. 7. Results from YOLOv8l + DeepSORT.

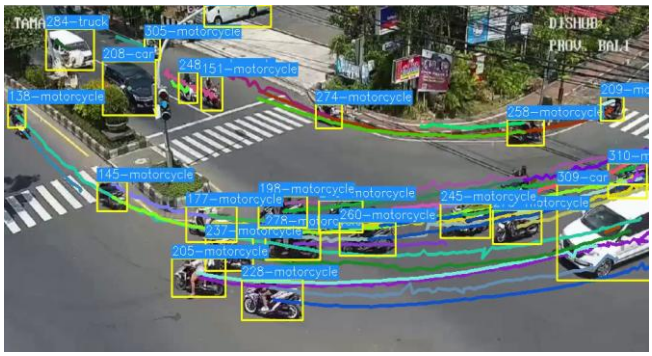


Fig. 8. Results from YOLOv8x + DeepSORT.

We subsequently present comprehensive HOTA evaluation results for the car class utilising a combination of the nano (n) and extra-large (x) YOLOv8 variants as detectors, paired with DeepSORT for association, and underscore the importance of these findings in determining the optimal detector architecture. The comprehensive assessment findings for all object classes and YOLOv8 versions will be displayed in tabular style and subsequently analysed in the following section. The CLEAR measure outcomes for this and further settings will be offered to offer a comprehensive view of performance tracking.

Tables I and II present the HOTA tracking outcomes for the "car" and "motorcycle" categories, respectively. According to the statistics in Tables I and II, a consistent trend is evident across all YOLOv8 model variants (from n to x): the overall tracking system exhibits marginally better and steadier performance for the car class than for the motorcycle class. The discrepancies in the primary HOTA scores are minimal; for instance, the top-performing x variant recorded a HOTA of 58.717, whereas the motorcycle attained 57.781. However, a thorough examination of the sub-metrics uncovers the origins of these performance variances and underscores the difficulties associated with motorcycle tracking.

TABLE I. HOTA TABLE FOR CAR CLASS

	Sub Metric	Yolov8 variant				
		n	*s*	*m*	*l*	*x*
	HOTA	53.137	55.541	55.596	58.477	58.717
SUB METRIC HOTA	DetA	47.491	50.915	52.408	55.602	54.479
	AssA	59.696	61.066	59.576	62.116	64.030
	DetRe	56.101	58.961	59.166	62.836	60.942
	DetPr	59.958	60.84	62.511	63.027	64.160
	AssRe	65.526	67.383	65.626	68.539	70.243
	AssPr	69.216	68.426	68.554	68.847	70.293
	LocA	74.073	73.390	73.131	73.298	73.746
	OWTA	57.869	59.970	59.304	62.415	62.379
	HOTA(0)	87.410	94.123	94.755	98.984	96.915
	LocA(0)	63.555	62.342	62.11	62.627	63.598
	HOTALocA(0)	55.554	58.678	58.853	61.991	61.636

TABLE II. HOTA TABLE FOR MOTOR CYCLE CLASS

		Yolov8 variant				
		n	*s*	*m*	*l*	*x*
	HOTA	51.545	54.140	55.372	57.703	57.781
SUB METRIC HOTA	DetA	47.820	48.684	53.002	55.624	53.842
	AssA	55.736	60.526	57.997	60.031	62.190
	DetRe	53.732	53.217	58.477	61.239	59.855
	DetPr	64.708	64.193	64.742	66.204	65.295
	AssRe	60.219	65.965	63.478	65.114	67.044
	AssPr	70.795	68.090	67.685	69.561	70.067
	LocA	75.339	73.128	73.620	74.365	74.317
	OWTA	54.656	56.696	58.198	60.611	60.993
	HOTA(0)	79.843	89.727	92.133	92.434	91.351
	LocA(0)	67.250	63.485	63.996	65.705	65.968
	HOTALocA(0)	53.694	56.963	58.962	60.734	60.262

Disaggregating HOTA results into Detection Accuracy (DetA) and Association Accuracy (AssA) provides an essential initial perspective. In these metrics, automobiles routinely surpass motorcycles. The performance disparity is more evident in AssA (e.g. 64,030 for cars against 62,190 for motorbikes in variation x) than in DetA (54,479 versus 53,842). This indicates that although modest difficulties exist in recognising motorcycles, a more substantial problem pertains to the tracker's capacity to sustain consistent motorbike IDs over time. The tracker misidentifies or erroneously exchanges motorbike IDs more frequently than it does for automobiles. Additional examination of the associated sub-metrics corroborates this conclusion. The primary distinction is evident in Association Recall (AssRe), with automobiles routinely achieving superior scores (e.g. 70,243 compared to 67,044 for variation x). Reduced AssRe scores for motorbikes immediately signify a track fragmentation problem. This indicates that a singular, ostensibly continuous motorbike route is often disrupted and recommenced as a new track with a distinct ID. This is probably because of the inherent characteristics of motorcycles: their reduced dimensions render them more vulnerable to complete obstruction by other vehicles, and their nimble and non-linear movement complicates motion prediction algorithms such as the Kalman Filter often employed in DeepSORT. Conversely, the AssPr (Association Precision) scores are almost indistinguishable across the two classes, suggesting that the challenge of consolidating several distinct objects into a single track (track merging) is not a significant differentiating factor.

The Detection Recall (DetRe) measure for motorbikes is marginally lower, indicating that the detector frequently overlooks motorcycles (false negatives) compared to vehicles, perhaps attributable to their reduced size. Notably, the Detection Precision (DetPr) for bikes is somewhat superior, suggesting that when the model identifies a motorbike, the forecast is generally more dependable (fewer false positives) compared to its predictions for cars. This HOTA measure research quantitatively demonstrates that the primary issue in tracking motorbikes, as opposed to cars, resides not merely in detection,

but predominantly in data association, particularly in preserving trajectory consistency despite agile motions and frequent occlusions.

In summary, in the automobile class, the mean HOTA and sub-metrics rose from YOLOv8-n (61.3) to YOLOv8-x (65.7), exhibiting a standard variation of around 10 points, which signifies a constant enhancement in performance with larger models. In the motorbike class, a same tendency is observed: YOLOv8-n (61.3) enhances to YOLOv8-l/x (65.7); however, the elevated standard deviation suggests considerable variability among sub-metrics. Elevated maximum values (e.g. >90 for HOTA(0)) suggest that certain measurements demonstrate substantial performance improvements with larger YOLOv8 variations.

The CLEAR metric analysis offers a compelling narrative that seems contradicts the HOTA results. Table III is the result of CLEAR metric measurements for the car class and Table IV is the result of CLEAR metric measurements for the motorcycles class. Remarkably, the Multiple Object Tracking Accuracy (MOTA) measure consistently surpasses that of vehicles for motorcyclists across all iterations of the YOLOv8 model. In the most sophisticated model x, motorcycles attain a MOTA of 62,622, whereas vehicles achieve only 57,331. This discovery necessitates a thorough examination of the error components constituting the MOTA score to comprehend the fundamental dynamics.

The explanation for the elevated MOTA ratings for motorcycles is found in the analysis of the precision and recall metrics derived by CLEAR. The data indicate that tracking precision (CLR_Pr) for bikes markedly exceeds that of automobiles across all model variants (e.g. 84,199 for motorcycles compared to 80,179 for cars in variant x). MOTA is determined by the aggregate of errors, including False Positives (FP), False Negatives (FN), and ID Switches (IDS), and exhibits significant sensitivity to the quantities of FP and FN. Increased precision immediately indicates a markedly reduced quantity of.

TABLE III. CLEAR METRIC TABLE FOR CAR CLASS

Sub Metric	Yolov8 variant				
	n	*s*	*m*	*l*	*x*
MOTA	44.415	47.904	50.888	56.604	57.331
MOTP	70.630	69.671	69.418	69.374	69.790
MODA	44.894	47.904	50.888	56.680	57.331
CLR_Re	69.231	72.408	72.768	78.188	76.158
CLR_Pr	73.990	74.715	76.882	78.426	80.179
MTR	19.737	28.571	31.746	41.935	34.722
PTR	48.684	35.714	20.635	29.032	45.833
MLR	31.579	35.714	47.619	29.032	19.444
IDSW	24	16	8	5	5
Frag	212	536	547	515	434

False Positives. Consequently, although motorbikes may encounter a higher frequency of association mistakes, a diminished number of false positives significantly decreases the overall errors in the MOTA formula, thereby enhancing their score relative to automobiles. A higher MOTA does not inherently indicate superior tracking performance; instead, it may signify the detector's exceptional accuracy for that specific class.

TABLE IV. CLEAR METRIC TABLE FOR MOTOR CYCLE CLASS

Sub Metric	Yolov8 variant				
	n	*s*	*m*	*l*	*x*
MOTA	53.906	51.935	55.943	63.886	62.622
MOTP	70.921	68.812	69.581	70.169	70.076
MODA	55.213	52.311	56.236	63.953	62.699
CLR_Re	69.125	67.606	73.279	78.227	77.183
CLR_Pr	83.246	81.550	81.131	84.569	84.199
MTR	31.169	44.495	28.571	30.435	40.370
PTR	58.442	37.615	44.643	58.261	42.609
MLR	10.390	17.890	26.786	11.304	16.522
IDSW	42	20	17	4	5
Frag	342	554	702	693	532

MOTA indicates a benefit for motorcycles; nevertheless, measures emphasising trajectory consistency present a contrasting narrative, aligning more closely with the prior HOTA findings. The quantity of FRAGs (Fragmentations) and IDSW (ID Switches) is continually elevated for motorcycles. In version x, motorcyclists exhibit 532 fragmentations, whereas vehicles have 434. In a less robust model, such as n, motorcyclists possess 42 IDSWs, whereas cars have just 24. These figures quantitatively validate that motorbike routes are more often fractured and their identities are more frequently exchanged. This substantiates the conclusion derived from the HOTA investigation that data association—sustaining consistent IDs amidst occlusion and dynamic movement, poses a far bigger problem for motorcycles.

Alternative trajectory quality measurements offer a more thorough perspective. The Mostly Tracked Ratio (MTR) for bikes is consistently superior (40,370 compared to 34,722 in the x variation); however, the Mostly Lost Ratio (MLR) is inferior (16,522 versus 19,444). This indicates that although motorcycle trajectories are prone to intermittent fragmentation, a greater percentage of the overall trajectories are effectively monitored during the majority of their duration. This suggests that your tracker is proficient at managing motorcycles in straightforward movement situations, although it struggles during complex events (sharp manoeuvres, significant occlusions) that result in IDSW and Frag. The nearly similar MOTP (Multiple Object Tracking Precision) ratings for both categories suggest that when an object is accurately recognised and appropriately associated, the bounding box localisation accuracy is equally proficient for both automobiles and motorcycles.

The results for motorbike tracking demonstrate that YOLOv8-x attains a HOTA score of 57.8, much surpassing the 51.5 achieved by YOLOv8-n, hence validating that larger detector variants enhance detection recall and association stability for tiny objects. Nonetheless, the aggregate HOTA values for motorbikes are inferior to those of cars, mostly attributable to recurrent occlusions, elevated object density, and the limited visual footprint of motorcycles in traffic scenarios. This finding aligns with the research of Frank Ngeni et al. [7], who indicated that tracking performance for small-scale and rapidly moving objects is susceptible to identification fragmentation, despite the utilisation of sophisticated detectors. Likewise, Jorge E. Espinosa1 et al. [4] noted that occlusion and swift manoeuvres markedly elevate ID switches in multi-object tracking benchmarks, underscoring the difficulties faced in this work. The comparisons indicate that although the integration of YOLOv8 with DeepSORT enhances motorcycle tracking relative to smaller detector variants, persistent challenges such as occlusion and identity consistency remain unaddressed, implying that future research should incorporate appearance-based re-identification modules or alternative tracking methods.

V. Conclusion

This article presents numerous major conclusions regarding the efficacy of a vehicle tracking system utilising the YOLOv8 detector in conjunction with the DeepSORT tracker, based on a thorough analysis undertaken. There exists a distinct positive correlation between the size of the YOLOv8 model and its detection accuracy; larger variants, such as YOLOv8l and YOLOv8x, markedly decrease the incidence of missed detections (false negatives), particularly for smaller objects like motorcycles, although difficulties in accurately classifying ambiguously shaped vehicles persist. Secondly, the dual-metric analysis provides significant complementary insights: the HOTA measure demonstrates that car tracking is more stable and consistent (higher "AssA" scores) compared to motorcycles, which frequently encounter track fragmentation due to their diminutive size, occlusion, and agile manoeuvres. Secondly, conversely, the CLEAR MOTA metric unexpectedly indicates elevated scores for motorcycles. This improvement is not attributable to enhanced association but rather to increased detection precision (fewer false positives), which disproportionately elevates the MOTA score and conceals deficiencies in tracking consistency (more ID shifts and fragmentation). This discovery significantly corroborates the original premise that a detection-biased statistic such as MOTA may yield an inadequate representation of actual tracking performance. This study indicates that HOTA offers a more equitable and comprehensive assessment by individually evaluating detection and association quality. This indicates that the selection of detector architecture and assessment metrics must be customised to the specific application requirements: in safety-critical systems, where identification consistency is essential, prioritising the HOTA association score is vital.

Subsequent study may focus on investigating the integration of YOLOv8 with different tracking algorithms beyond DeepSORT to enhance identification consistency and minimize trajectory fragmentation in multi-object tracking. A notable possibility is ByteTrack, which has a distinctive technique that accounts for low-rank detection boxes. Furthermore, advanced trackers like StrongSORT and OC-SORT merit assessment due of their enhancements in association consistency and tracking robustness in intricate traffic scenarios. Through the examination of diverse combinations, it is anticipated that a YOLOv8-based vehicle tracking system will attain enhanced performance and adaptability to a range of real-world conditions.

References

[1] M. Elassy, M. Al-Hattab, M. Takruri, and S. Badawi, "Intelligent transportation systems for sustainable smart cities," Transportation Engineering, vol. 16, p. 100252, Jun. 2024, doi: 10.1016/j.treng.2024.100252.

[2] L. Fei and B. Han, "Multi-Object Multi-Camera Tracking Based on Deep Learning for Intelligent Transportation: A Review," Sensors, vol. 23, no. 8, p. 3852, Apr. 2023, doi: 10.3390/s23083852.

[3] R. C. R. Nampalli, "Leveraging AI in Urban Traffic Management: Addressing Congestion and Traffic Flow with Intelligent Systems," Journal of Artificial Intelligence and Big Data, vol. 1, no. 1, pp. 86–99, May 2021, doi: 10.31586/jaibd.2021.1151.

[4] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Detection and Tracking of Motorcycles in Congested Urban Environments Using Deep Learning and Markov Decision Processes," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11524 LNCS, Springer Verlag, 2019, pp. 139–148. doi: 10.1007/978-3-030-21077-9_13.

[5] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," EURASIP J Image Video Process, vol. 2008, pp. 1–10, 2008, doi: 10.1155/2008/246309.

[6] J. Luiten et al., "HOTA: A Higher Order Metric for Evaluating Multi-object Tracking," Int J Comput Vis, vol. 129, no. 2, pp. 548–578, Feb. 2021, doi: 10.1007/s11263-020-01375-2.

[7] F. Ngeni, J. Mwakalonge, and S. Siuhi, "Solving traffic data occlusion problems in computer vision algorithms using DeepSORT and quantum computing," Journal of Traffic and Transportation Engineering (English Edition), vol. 11, no. 1, pp. 1–15, Feb. 2024, doi: 10.1016/j.jtte.2023.05.006.

[8] X. Hou, Y. Wang, and L.-P. Chau, "Vehicle Tracking Using Deep SORT with Low Confidence Track Filtering," in 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, Sep. 2019, pp. 1–6. doi: 10.1109/AVSS.2019.8909903.

[9] T.-N. Doan and M.-T. Truong, "Real-time vehicle detection and counting based on YOLO and DeepSORT," in 2020 12th International Conference on Knowledge and Systems Engineering (KSE), IEEE, Nov. 2020, pp. 67–72. doi: 10.1109/KSE50997.2020.9287483.

[10] L. Lin, H. He, Z. Xu, and D. Wu, "Realtime Vehicle Tracking Method Based on YOLOv5 + DeepSORT," Comput Intell Neurosci, vol. 2023, no. 1, Jan. 2023, doi: 10.1155/2023/7974201.

[11] D. Guo, Z. Li, H. Shuai, and F. Zhou, "Multi-Target Vehicle Tracking Algorithm Based on Improved DeepSORT," Sensors, vol. 24, no. 21, p. 7014, Oct. 2024, doi: 10.3390/s24217014.

[12] O. Angah and A. Y. Chen, "Tracking multiple construction workers through deep learning and the gradient based method with re-matching based on multi-object tracking accuracy," Autom Constr, vol. 119, p. 103308, Nov. 2020, doi: 10.1016/j.autcon.2020.103308.

[13] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu, "Improving Multiple Object Tracking with Single Object Tracking," in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jun. 2021, pp. 2453–2462. doi: 10.1109/CVPR46437.2021.00248.

[14] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim, "Multiple object tracking: A literature review," Artif Intell, vol. 293, p. 103448, Apr. 2021, doi: 10.1016/j.artint.2020.103448.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[16] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[17] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 2018, [Online]. Available: http://arxiv.org/abs/1804.02767

[18] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[19] B. Mahaur and K. K. Mishra, "Small-object detection based on YOLOv5 in autonomous driving systems," Pattern Recognit Lett, vol. 168, pp. 115–122, 2023, doi: https://doi.org/10.1016/j.patrec.2023.03.009.

[20] C. Li et al., "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," Sep. 2022, [Online]. Available: http://arxiv.org/abs/2209.02976

[21] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Jun. 2023, pp. 7464–7475. doi: 10.1109/CVPR52729.2023.00721.

[22] M. Sohan, T. Sai Ram, and Ch. V. Rami Reddy, "A Review on YOLOv8 and Its Advancements," in Data Intelligence and Cognitive Informatics, S. and F.-G. P. Jacob I. Jeena and Piramuthu, Ed., Singapore: Springer Nature Singapore, 2024, pp. 529–545. doi: 10.1007/978-981-99-7962-2_39.

[23] M. Safaldin, N. Zaghden, and M. Mejdoub, "An Improved YOLOv8 to Detect Moving Objects," IEEE Access, vol. 12, pp. 59782–59806, 2024, doi: 10.1109/ACCESS.2024.3393835.

[24] I. N. E. Indrayana, M. Sudarma, I. K. G. D. Putra, and A. A. K. O. Sudana, "Improve Nighttime Highway Vehicles and Pedestrian Detection Using Yolov8+CLAHE," in 2024 Ninth International Conference on Informatics and Computing (ICIC), IEEE, Oct. 2024, pp. 1–6. doi: 10.1109/ICIC64337.2024.10956682.

[25] I. Purwita Sary, E. Ucok Armin, S. Andromeda, E. Engineering, and U. Singaperbangsa Karawang, "Performance Comparison of YOLOv5 and YOLOv8 Architectures in Human Detection Using Aerial Images," Ultima Computing : Jurnal Sistem Komputer, vol. 15, no. 1, 2023.

[26] H. K. Jooshin, M. Nangir, and H. Seyedarabi, "Inception - YOLO: Computational cost and accuracy improvement of the YOLOv5 model based on employing modified CSP, SPPF, and inception modules," IET Image Process, vol. 18, no. 8, pp. 1985 – 1999, Jun. 2024, doi: 10.1049/ipr2.13077.

[27] A. N. Alhawsawi, S. D. Khan, and F. U. Rehman, "Enhanced YOLOv8-Based Model with Context Enrichment Module for Crowd Counting in Complex Drone Imagery," Remote Sens (Basel), vol. 16, no. 22, p. 4175, Nov. 2024, doi: 10.3390/rs16224175.

[28] W. Yang, X. Tang, K. Jiang, Y. Fu, and X. Zhang, "An Improved YOLOv5 Algorithm for Vulnerable Road User Detection," Sensors, vol. 23, no. 18, p. 7761, Sep. 2023, doi: 10.3390/s23187761.

[29] M. I. H. Azhar, F. H. K. Zaman, N. Md. Tahir, and H. Hashim, "People Tracking System Using DeepSORT," in 2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), IEEE, Aug. 2020, pp. 137–141. doi: 10.1109/ICCSCE50387.2020.9204956.

[30] D. M. Jiménez-Bravo, Á. Lozano Murciego, A. Sales Mendes, H. Sánchez San Blás, and J. Bajo, "Multi-object tracking in traffic environments: A systematic literature review," Neurocomputing, vol. 494, pp. 43–55, Jul. 2022, doi: 10.1016/j.neucom.2022.04.087.

[31] E. Bayraktar, "Advanced Kalman Filter Optimization for Efficient Multi-Object Tracking in Computer Vision," in 2024 Innovations in Intelligent Systems and Applications Conference (ASYU), IEEE, Oct. 2024, pp. 1–6. doi: 10.1109/ASYU62119.2024.10757018.

[32] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in 2017 IEEE International Conference on Image Processing (ICIP), IEEE, Sep. 2017, pp. 3645–3649. doi: 10.1109/ICIP.2017.8296962.

[33] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," EURASIP J Image Video Process, vol. 2008, pp. 1–10, 2008, doi: 10.1155/2008/246309.

[34] A. B. Holla, M. M. M. Pai, U. Verma, and R. M. Pai, "Vehicle Re-Identification and Tracking: Algorithmic Approach, Challenges and Future Directions," IEEE Open Journal of Intelligent Transportation Systems, vol. 6, pp. 155–183, 2025, doi: 10.1109/OJITS.2025.3538037.

[35] L. Sharan, H. Kelm, G. Romano, M. Karck, R. De Simone, and S. Engelhardt, "mvHOTA: A multi-view higher order tracking accuracy metric to measure temporal and spatial associations in multi-point tracking," Comput Methods Biomech Biomed Eng Imaging Vis, vol. 11, no. 4, pp. 1281–1289, Jul. 2023, doi: 10.1080/21681163.2022.2159535.

[36] V. Nayak, B. Sonar, N. Marali, K. Mallibhat, and P. Nissimagoudar, "Comparative Analysis of Neural Trajectory Prediction Models for Vehicle Tracking," in Proceedings of 5th International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications, J. M. Gunjan Vinit Kumar and Zurada, Ed., Singapore: Springer Nature Singapore, 2025, pp. 37–50. doi: 10.1007/978-981-97-8865-1_4.

[37] E. Syahrudin, E. Utami, and A. D. Hartanto, "Object Detection with YOLOv8 and Enhanced Distance Estimation Using OpenCV for Visually Impaired Accessibility," JOIV : International Journal on Informatics Visualization, vol. 9, no. 2, p. 575, Mar. 2025, doi: 10.62527/joiv.9.2.2826.

[38] S. Pudaruth, I. M. Boodhun, and C. W. Onn, "Reducing Traffic Congestion Using Real-Time Traffic Monitoring with YOLOv8," International Journal of Advanced Computer Science and Applications, vol. 15, no. 10, 2024, doi: 10.14569/IJACSA.2024.01510109.

[39] R. Kridalukmana, D. Eridani, R. Septiana, and I. P. Windasari, "YoloV8, EfficientNetv2, and CSP Darknet Comparison as Recognition Model's Backbone for Drone-Captured Images," JOIV : International Journal on Informatics Visualization, vol. 9, no. 2, p. 683, Mar. 2025, doi: 10.62527/joiv.9.2.2880.

[40] Y. Li et al., "Beyond MOT: Semantic Multi-object Tracking," 2025, pp. 276–293. doi: 10.1007/978-3-031-72761-0_16.

[41] S. V. Ganesh, Y. Wu, G. Liu, R. Kompella, and L. Liu, "Amplifying Object Tracking Performance on Edge Devices," in 2023 IEEE 5th International Conference on Cognitive Machine Intelligence (CogMI), IEEE, Nov. 2023, pp. 83–92. doi: 10.1109/CogMI58952.2023.00021.

Sentence-Level Indonesian Sign Language (BISINDO) Recognition Using 3D CNN-LSTM and 3D CNN-BiLSTM Models

Katriel Larissa Wiguna, Rojali
Computer Science Department-BINUS Graduate Program–Master of Computer Science,
Bina Nusantara University, Jakarta, Indonesia, 11480

***Abstract*—Sign Language Recognition (SLR) has been an active area of research, but sentence-level SLR remains relatively underexplored. While most studies focus on recognizing individual signs, understanding full sentences presents greater challenges. This research proposes a sentence-level SLR using a combination of 3D Convolutional Neural Networks (3D CNN) for spatio-temporal feature extraction with sequential modeling using Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM). Connectionist Temporal Classification (CTC) is also used to enable training without word-level annotations. In this study, we used the Indonesian Sign Language (BISINDO) dataset, specifically the DKI Jakarta version, consisting of 900 videos representing 30 sentences, which was expanded to 3600 videos through data augmentation techniques such as speed variation and brightness adjustments. All videos underwent preprocessing to ensure data quality, and Bayesian Optimization was applied for hyperparameter tuning to obtain optimal configurations for each model. Both models were trained with CTC loss and evaluated using Word Error Rate (WER). The 3DCNN-LSTM model achieved a WER result of 59.21%, while the 3DCNN-BiLSTM presents a significantly better performance with a WER of 2.77%. Despite these promising results, the models' ability to generalize across different signers may require further research, as the dataset used in this research involved only a single signer.**

Keywords*—Sign Language Recognition; BISINDO (Indonesian Sign Language); 3D Convolutional Neural Network (3D CNN); Long Short-Term Memory (LSTM) Network; Bidirectional Long Short-Term Memory (BiLSTM); Connectionist Temporal Classification (CTC)

I. Introduction

As social creatures, communication is an essential part that is inseparable from human life. Through communication, humans are able to exchange ideas, build relationships, and collaborate to fulfill their needs and achieve their goals. Broadly speaking, communication can be classified into verbal communication, which uses spoken or written words, and non-verbal communication, which relies on facial expressions, gestures, and body language.

Despite the fact that verbal communication, especially speaking, is the most common way humans interact, it is undeniable that some people cannot speak. Deaf individuals, unlike the general population, are among those who are unable to speak. For this reason, the deaf community communicates in their own, unique way, which is through the use of sign language.

In Indonesia, there are two types of sign language used: Indonesian Sign Language System (SIBI) and Indonesian Sign Language (BISINDO). SIBI is a sign language system that follows standard Indonesian grammar, which includes basic signs that symbolizes words, and additional signs for prefixes, suffixes, and particles. However, it is not fully accepted by the deaf community, who find it difficult for daily use. Although SIBI is used in formal education, BISINDO is more commonly used by the deaf for everyday communication. Research at PGMI UIN Sunan Kalijaga shows BISINDO is easier to understand, more natural, expressive, and effective than SIBI [1].

The difference in communication styles between the deaf and hearing communities creates a communication gap, which can lead to social distance and discrimination within society. Therefore, a system is needed to help bridge this gap. As a solution to this gap, sign language recognition (SLR) systems have been widely developed [2]. The goal of SLR is to develop a method capable of detecting and understanding the sign language gestures made by individuals [2]. Various models have been applied in the development of sign language recognition systems, ranging from Convolutional Neural Networks (CNN) [3], Bidirectional Recurrent Neural Networks [4], Transformers [5], Long Short-Term Memory Networks (LSTM) [6], deep convolutional networks [7], to a combination of 2 or more models [8].

Considering the fact that Indonesia is one of the four countries in Asia with a fairly high prevalence of hearing loss at 4.6%, and a deafness prevalence of 0.4% across all age groups in seven provinces, a continuous sign language recognition, also known as sentence-level SLR, is very feasible to be developed to help deaf individuals communicate with hearing individuals. While this system helps translate what deaf individuals want to convey to hearing individuals, it also serves as a tool for hearing individuals to learn sign language. In Indonesia itself, there has been a research on sentence-level SLR using 3D CNN and Bidirectional Recurrent Neural Networks (BiRNN) [4]. However, the model developed utilizes the SIBI dataset, which is not frequently used by deaf individuals in their daily interactions. Besides, it shows a relatively high word error rates (WER) and character error rates (CER), which may be caused by a mismatch between the

model and the dataset used, such as the presence of noise in the dataset due to suboptimal preprocessing. Additionally, sentence-level SLR is generally more challenging than alphabet or word-level recognition due to the need to capture both manual and non-manual features, while also modeling temporal dependencies across multiple gestures and handling variations in gesture sequences.

In response to this problem, this study proposes a sentence-level SLR model for Indonesian Sign Language (BISINDO) using 3D CNN-LSTM and 3D CNN-BiLSTM models. To address limitations in previous studies, the dataset consists of BISINDO videos collected with consistent lighting, framing, and other conditions to minimize noise. LSTM and BiLSTM are used instead of BiRNN due to their gating mechanisms, which better handle vanishing gradients and capture long-term dependencies in gesture sequences [9].

Since BISINDO naturally develops within the Deaf community, there are lots of variations in BISINDO gestures across different regions, reflecting the local culture in each area. This results in several regional variants of BISINDO, such as BISINDO DKI Jakarta, BISINDO West Java, BISINDO East Java, and others. However, since DKI Jakarta is the province with the highest population density in Indonesia [10] and as a central area where many communities interact, the DKI Jakarta version of BISINDO is likely to be more widely used and recognized than variants of other regions. Therefore, we chose to focus on BISINDO DKI Jakarta in this research.

To sum up, this study aims to develop and evaluate a sentence-level BISINDO DKI Jakarta recognition using a combination 3D CNN and LSTM model, as well as 3D CNN and BiLSTM model. While previous studies have focused on word or alphabet-level recognition, only a few have focused on sentence-level SLR. This gap motivates the development of a model capable of recognizing sentence-level sign language gestures. The proposed model is expected to provide a basis for developing future systems that support communication between Deaf and hearing communities. It has the potential to support sign language learning, contribute to the development of applications such as sign language translators, and enable future innovations in automated public service systems.

To address these objectives, this study seeks to answer the following research questions: 1) How can hybrid 3D CNN-LSTM and 3D CNN-BiLSTM models be developed to effectively recognize sentence-level BISINDO gestures? 2) How do the performance of the 3D CNN-LSTM and 3D CNN-BiLSTM models compare in terms of their accuracy in recognizing word sequences within BISINDO sentences?

The next section of this paper is organized as follows: Section II presents related works on Sign Language Recognition, Section III presents the methods used in this research. The results and discussion are presented in Section IV and Section V, respectively. Finally, Section VI presents the conclusion and future works of this research.

II. Related Works

To date, there has been considerable research focused on sign language recognition systems. Different algorithms have been implemented by previous researchers to develop a sign language recognition system that can perform with good accuracy.

In 2018, Ariesta et al. proposed a sentence-level Indonesian sign language recognition system using 3D CNN and Bidirectional Recurrent Neural Networks. The study utilized SIBI dataset, which includes 30 sentences in SIBI. The researchers developed several models for this system, all combining 3D CNN and bidirectional RNN. However, the evaluation revealed that the performance of the developed models was not optimal, as indicated by the high Word Error Rate (WER) of 85%-90% and Character Error Rate (CER) of 65%-77% [4]. This study is notable for attempting sentence-level sign language recognition, which is a challenging task. The use of 3D CNN-BiRNN is also suitable for capturing spatio-temporal features. However, the dataset and the model seem to be mismatched, resulting in suboptimal performance. This highlights the importance of preparing the dataset carefully for sentence-level recognition.

Aljabar and Suharjito in 2020 proposed a word-level sign language recognition system, utilizing BISINDO dataset consisting of two alphabets and eight words in BISINDO. They developed CNN, LSTM, and a combination of CNN and LSTM model, in which the CNN model achieves an accuracy of 73%, LSTM model achieves an accuracy of 81%, and the combination of CNN and LSTM model achieves the highest accuracy, reaching 90% [11].

Another research in 2020 by Latif et al. proposed an alphabet-level sign language recognition system to detect alphabet in Arabic sign language. They used a deep convolutional network, which was trained and evaluated using a dataset consisting of 50000 images of Arabic signs. After evaluation, it is found that this model achieves an accuracy of 97.6% [7]. While this study achieves a high accuracy in recognizing Arabic letters with its large and diverse dataset, it is still limited to letters rather than word or sentence-level recognition.

In the same year, a continuous sign language neural machine translation was proposed by a group of researchers from China. In this research, they used a dataset consisting of 50 sentences in Chinese Sign Language. The model proposed in this research is the ST-LSTM fusion attention network, which is then called Bi-ST-LSTM-A. This model achieves an accuracy of 81.22% on the CSL dataset, 76.12% on the RWTH-PHOENIX-Weather dataset, and 75.32% on the RWTH-PHOENIX-2014T dataset [8].

In 2021, a group of researchers from Saudi Arabia proposed an Arabic Sign Language recognition to detect alphabets in Arabic Sign Language using R-CNN, in which they achieved an accuracy of 93% based on the evaluation conducted [12]. Although this study focuses only on recognizing alphabets in Arabic Sign Language, the model proposed in this study can detect hands and recognize gestures with a high accuracy, even in a complicated background.

Another researcher from Indonesia also proposed a sign language detection system for BISINDO in 2022. They used a dataset of nine class, which consists of the alphabet A-E, and

three words which are “saya”, “kamu”, and “I love you”. They used CNN to detect those classes, in which they achieved an accuracy of 99.82% [13]. This study still needs a lot of improvements, especially in terms of the dataset size, as the limited vocabulary makes it unreliable in real-world scenarios.

Another alphabet-level sign language recognition is also proposed by Murali et al. in 2022. They used a CNN model to detect alphabets in American Sign Language, in which they got an accuracy of 98% [3].

In the same year, research by Kothadiya et al. proposed a system to detect words in Indian Sign Language. They used deep learning models, especially the Long Short-Term Memory (LSTM) and GRU. The model, which consists of 1 LSTM layer and GRU, achieves an accuracy of 97% for more than 11 word classes [14]. This shows a high accuracy for Indian Sign Language Recognition in a challenging and uncontrolled environment, representing natural conditions. However, the dataset is still limited in diversity, containing only 11 words, which may restrict the model’s ability to recognize sign language in real-world applications.

Another alphabet and word-level sign language recognition is proposed by Shin et al. in 2023. They combined a CNN and a transformer model to create a model to detect alphabet and words in the KSL Dataset. After evaluation, almost all alphabets are identified correctly, two alphabets achieved an accuracy of 95%, and two others gained an accuracy of 84%-90%. Testing was also done on the proposed dataset, which consists of 20 classes, achieving a score of 98.5% for precision, 98.35% for recall, 98.4% for F1-score, and 98.3% for accuracy [5]. This study effectively combines CNN and transformer architectures, resulting in high accuracy with relatively low computational cost. However, it is limited to word-level recognition and relies on a relatively small dataset, which limits its applicability to continuous, real-world sign language scenarios.

In 2023, a group of researchers from India proposed a system using the LSTM networks to detect words in Indian Sign Language. This model achieves an accuracy of 87% [6]. This study achieves a relatively high accuracy, and it is strong in its practical applicability, demonstrating a usable, real-time sign-action recognition system. However, it is limited to word-level recognition and does not address sentence-level sequences.

Lastly, in 2024, a hybrid deep learning model was proposed for a real-time Arabic Sign Language. This study presents a new, custom dataset containing 10 static gesture words and 10 dynamic gesture words. CNN and LSTM were used in this study, with the CNN achieving accuracy rates of 94.40% and the LSTM achieving 82.70% [15]. The results highlight the strong potential of the proposed model, especially proving CNN’s effectiveness in recognizing static signs. However, the dataset used in this research is relatively small, which makes it less applicable in real-world scenarios. Expanding the dataset in the future, especially for dynamic gestures, is essential to achieve a more reliable sign language recognition system.

From the previous research, it can be concluded that sign language recognition systems with various models, especially CNN and LSTM can be used to detect gestures in sign language with a fairly good level of accuracy, with an average accuracy of more than 90% for word-level sign language recognition. However, research related to sentence-level sign language recognition is still limited. Based on previous research that has been discussed, there are only two groups of researchers who studied sentence-level sign language recognition, where 1 of them shows a large error rate, and 1 the other shows fairly good accuracy results (75%- 82%), although not as good as the accuracy of the system in the word-level sign language recognition. Therefore, referring to previous research, this study will develop sentence-level sign language recognition using a combination of 3D CNN and LSTM models. This study was conducted to detect gestures in the DKI Jakarta version of BISINDO.

III. Methodology

This section presents the detailed research workflow, describing the step-by-step process involved in preparing the data, building and training the model, to evaluating its performance for BISINDO DKI Jakarta sentence-level SLR.

A. Problem Identification

In the problem identification phase, we focused on highlighting specific issues within the field of Sign Language Recognition (SLR), aiming to address gaps in current research. This process was done through an extensive review of recent studies and literature related to SLR. Even though numerous researchers have successfully developed SLR, most of them still focus on word-level Sign Language Recognition, so the communication gap between deaf and hearing individuals is not fully addressed.

In order to address the problem, this research focuses on developing a sentence-level SLR using the BISINDO dataset, a sign language which is preferred by deaf individuals in Indonesia. While the BISINDO dataset varies across Indonesia, this research focuses particularly on the DKI Jakarta version of BISINDO.

B. Video Acquisition

While several BISINDO datasets are available at the word or alphabet level, no publicly accessible dataset exists for sentence-level BISINDO (especially the DKI Jakarta version). Due to this gap, we created our own BISINDO DKI Jakarta dataset to be used in this research.

The video acquisition phase began with compiling a list of sentences to be recorded, which will then be used in this study. To construct the sentences, we utilizes vocabularies which are included in the 50 high frequency words based on research conducted by Siagian, which are recommended as mandatory vocabularies to be mastered when learning a new foreign language [16]. From this list, we selected several words to construct the sentences. The selection of these words was based on their frequency in everyday conversations and their applicability within Bahasa Isyarat Indonesia (BISINDO). While most words were selected from the 50 high-frequency words list, additional words outside the list were also incorporated as needed to form meaningful sentences. The sentences constructed from these words are then recorded to be

used as input for training and testing our sentence-level sign language recognition model. The list of sentences which will be included in this dataset can be seen in Table I. The "Sentence" column contains the original sentence in Indonesian, while the "Sign Gloss" column represents the translation of the sentence into BISINDO gloss notation. As shown in the examples, the gloss often differs from the spoken sentence structure, illustrating the linguistic differences between spoken Indonesian and BISINDO, which the recognition model must learn to accurately align sentences with their corresponding signs.

TABLE I. LIST OF SENTENCES IN THE DATASET

No.	Sentence	Sign Gloss
1.	Aku bangun jam sepuluh.	AKU BANGUN JAM SEPULUH
2.	Aku tidur jam sepuluh malam sampai jam lima pagi.	AKU MALAM TIDUR JAM SEPULUH SAMPAI PAGI JAM LIMA
3.	Aku mau makan.	AKU MAU MAKAN
4.	Aku punya kakak.	AKU PUNYA KAKAK
5.	Aku suka kamu.	AKU SUKA KAMU
6.	Di mana Bapak dan Ibu?	BAPAK DAN IBU DIMANA
7.	Halo, nama aku Katriel	HALO NAMA AKU FS:KATRIEL
8.	Ibu aku punya kucing dan ikan.	IBU AKU PUNYA KUCING DAN IKAN
9.	Kakak aku suka makan.	KAKAK AKU SUKA MAKAN
10.	Berapa anak yang kamu punya?	KAMU ANAK BERAPA
11.	Jam berapa kamu bangun?	KAMU BANGUN JAM BERAPA
12.	Apakah kamu sudah makan?	KAMU MAKAN SUDAH
13.	Keluarga aku terdiri dari lima orang.	KELUARGA AKU ORANG LIMA
14.	Berapa jumlah anggota keluarga kamu?	KELUARGA KAMU ORANG BERAPA
15.	Kucing makan apa?	KUCING MAKAN APA
16.	Kucing makan ikan.	KUCING MAKAN IKAN
17.	Siapa nama dia?	NAMA DIA SIAPA
18.	Siapa nama kamu?	NAMA KAMU SIAPA
19.	Nama isyarat kamu apa?	NAMA PANGGIL KAMU APA
20.	Rumah aku di Jakarta	RUMAH AKU FS:JAKARTA
21.	Rumah aku nomor lima.	RUMAH AKU NOMOR LIMA
22.	Rumah kamu di mana?	RUMAH KAMU DIMANA
23.	Rumah kamu nomor berapa?	RUMAH KAMU NOMOR BERAPA
24.	Sekarang jam berapa?	SEKARANG JAM BERAPA
25.	Selamat hari Ibu.	SELAMAT HARI IBU
26.	Selamat hari kemerdekaan Indonesia.	SELAMAT HARI KEMERDEKAAN INDONESIA
27.	Selamat malam, aku mau tidur.	SELAMAT MALAM AKU MAU TIDUR
28.	Selamat pagi, Bapak dan Ibu.	SELAMAT PAGI BAPAK DAN IBU
29.	Terima kasih	TERIMAKASIH
30.	Umur kamu berapa?	UMUR KAMU BERAPA

The recordings were carried out by a single participant – the author of this research – who is a non-native signer but has completed a sign language course, which provides sufficient experience to ensure accurate movements and expressions for the dataset. The decision to record with one signer was made due to limited resources, which made it challenging to involve multiple native signers. However, the recordings were carefully done to maintain consistency and clarity.

In total, 30 BISINDO sentences (DKI Jakarta version) were recorded, each repeated 30 times, using a smartphone camera at 30 fps. The video recording was carried out under controlled conditions, such as:

- Videos were recorded with a plain white background to minimize unnecessary visual distractions, to help the model focus on the signer.
- Videos were recorded indoors under sufficient lighting, ensuring every gesture was clearly visible.
- The camera was positioned to capture the signer's upper body (from head to waist), so that both hand movements and relevant facial expressions were clearly captured.

Nevertheless, relying on a single non-native signer remains a key challenge of this dataset, as it may limit the model's ability to generalize across different signers. To partially address this challenge, data augmentation techniques were applied to enhance the variability of the dataset, which will be described in the following section.

C. Video Preprocessing

To ensure that the recorded videos could be effectively used for model training, we applied several preprocessing steps. Firstly, we cropped the recorded videos to remove unnecessary parts of the frame, allowing the model to better focus on the signer's gestures and expressions. It is ensured that all videos have a resolution of 1080x1080 pixels. Next, the video frame rate was reduced from 30 fps to 15 fps. This was done to reduce computational load without changing the duration of the video. By reducing the fps, the total number of frames in each video can be cut in half while preserving necessary temporal information. Then, each frames are resized from 1080x1080 pixels to 256x256 pixels to maintain consistency in the input data, while also improving computational efficiency without losing essential details such as hand and facial features.

Next, padding was applied to the videos to ensure that the number of frames across all videos is the same. Since this research is focused on sentence-level sign language recognition, the number of frames of each video in the dataset will be different, due to the difference in sentence length or the variability in signing speed. Hence, padding was done by determining the maximum number of frames in a video across all videos in the dataset. The videos with fewer frames than the maximum number of frames are then given additional frames to equalize the number of frames across all videos in the dataset.

D. Data Augmentation

Since the total number of videos collected for this dataset is relatively small, consisting of only 900 videos covering 30 sentences, several data augmentation techniques were applied to enhance the variability of our training data. We carefully selected the augmentation techniques based on their compatibility with BISINDO, and made sure that the modifications did not alter the meaning of signs or even violate BISINDO rules. There are two augmentation techniques used in this research, which are:

1) Speed variation: This technique was used to increase the speed of the video by 1.2 times, which is intended to mimic the variability of signing speeds in real-world scenarios. It allows the model to learn and manage different signing speeds.

2) Brightness adjustments: This technique was used to modify the lighting conditions of the videos to simulate different lighting conditions found in different environments, which has been proven to improve model performance on object detection tasks [17]. In this study, brightness adjustments were applied by making the videos 40% darker and 40% brighter than the original video. Examples of brightness variations can be seen in Fig. 1 and Fig. 2. For privacy reasons, faces are covered in the dataset samples shown.



Fig. 1. Original video brightness.



Fig. 2. Video with 40% brighter lighting (left) and video with 40% darker lighting (right).

These techniques were used because they align with the characteristics of the dataset used. Other methods, such as mirroring, were avoided due to BISINDO rules on right and left hand usage, which could alter sign meaning. Other techniques like cropping or adding noise were also not applied, as they might remove essential facial expressions, hand positions, and other details needed for sign recognition.

E. Data Preparation

After all the preprocessing steps have been finished, the dataset was split into three groups, which are: training, validation, and testing sets. The data distribution can be seen in Table II.

TABLE II. ORIGINAL DATA DISTRIBUTION

Data Sets	Percentage	Number of Videos
Training Set	70%	630
Validation Set	15%	135
Testing Set	15%	135
TOTAL	100%	900

As shown in Table II, a 70-15-15 split was applied to the original dataset consisting of 900 videos. Meanwhile, the 2700 augmented videos were included in the training set, as the purpose of augmentation is to enrich the diversity of data available for training, allowing the model to generalize better. The final distribution dataset can be seen in Table III.

TABLE III. FINAL DATA DISTRIBUTION

Data Sets	Number of Videos
Training Set	3330
Validation Set	135
Testing Set	135
TOTAL	3600

F. Model Building and Training

In this phase, we developed a hybrid deep learning model that integrates a 3D CNN and recurrent model (LSTM/BiLSTM) for sentence-level sign language recognition. Since recognizing sign language requires attention and consideration to both manual features (hand shape, position, and movements) and non-manual features (facial expressions, head position, and lip shape) [18], the 3D CNN was utilized to extract spatial and temporal features from the video frames. These extracted features are then passed into two different sequence models, the conventional LSTM and bidirectional LSTM. This model design was chosen to leverage the main strengths of each model, where the 3D CNN serves as the feature extractor, while the LSTM/BiLSTM models capture temporal dependencies within the sign sequence. The model architecture used in this research is presented in Fig. 3, while the configuration of each layer is shown in Table IV. The rows highlighted in blue represent the 3D CNN component of the model, while the rows highlighted in orange represent the LSTM/BiLSTM component. All models were trained from scratch on the BISINDO DKI Jakarta dataset without the use of pre-trained weights or fine-tuning. This was done to ensure that the models are fully adapted to BISINDO sentence-level recognition data.

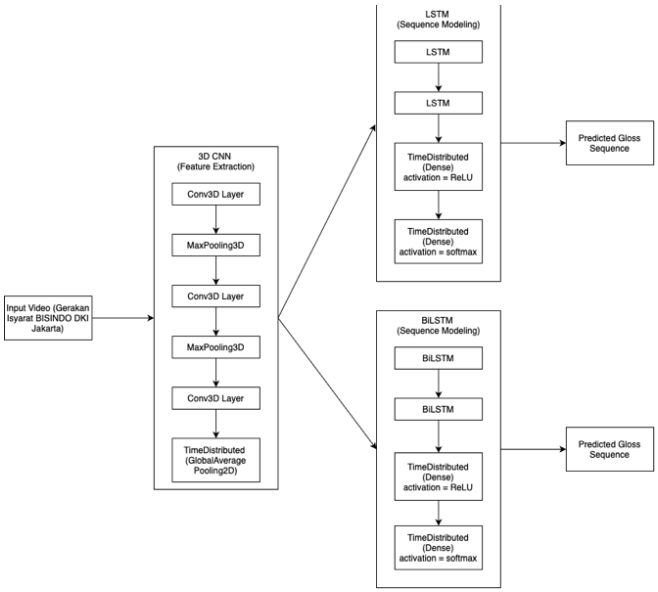


Fig. 3. Model architecture.

TABLE IV. LAYER CONFIGURATION OF THE 3D CNN AND LSTM/BILSTM MODEL

Layer	Configuration	Output Shape
Input 3D CNN	Video frames	(151, 256, 256, 3)
Conv3D	32 filters, kernel (3,3,3), stride (1,1,1), activation ReLU	(151, 256, 256, 3)
MaxPooling3D	Pool size (1, 2, 2)	(151, 128, 128, 32)
Conv3D	64 filters, kernel (3, 3, 3), stride (1, 1, 1), activation ReLU	(151, 128, 128, 64)
MaxPooling3D	Pool size (2, 2, 2)	(75, 64, 64, 64)
Conv3D	128 filters, kernel (3, 3, 3), stride (1, 1, 1), activation ReLU	(75, 64, 64, 128)
TimeDistributed (GlobalAveragePooling2D)		(75, 128)
Input LSTM	3D CNN extracted features	(75, 128)
LSTM/BiLSTM	X units (adjusted based on tuning results)	(75, X)
LSTM/BiLSTM	X units (adjusted based on tuning results)	(75, X)
TimeDistributed Dense (ReLU)	128 units, activation ReLU	(75, 128)
TimeDistributed Dense (Softmax)	45 units, activation Softmax	(75, 45)

As shown in Table IV, firstly the videos containing BISINDO sign language gestures were used as an input for the 3D CNN model. The 3D CNN then performed spatiotemporal feature extraction. The spatial features extracted include hand movements and facial expressions, as these two components are key elements in performing BISINDO, and are essential for identifying sign gestures.

The 3D CNN model used in this study receives video input with dimensions of (151, 256, 256, 3), where:

- 151 represents the number of frames in each video
- 256 x 256 is the resolution of each frame, and
- 3 represents the number of channels (RGB)

This input is processed through several convolutional and pooling layers as shown in Table IV. Then, a TimeDistributed layer with GlobalAveragePooling2D is applied to average the spatial feature values for each video frame individually, preserving the temporal dimension of the video data. This process results in a one-dimensional feature vector for each frame, which can then be passed into the LSTM/BiLSTM model. The output at this stage has the shape (75, 128), where:

- 75 is the number of frames remaining after pooling and downsampling, and
- 128 is the number of features generated for each frame.

To avoid repeating feature extractions due to limited computational resources, the extracted features were stored in a .npy file. These features are then used as an input to the LSTM/BiLSTM model. For sequence modeling, two stacked LSTM/BiLSTM layers were used to capture temporal dependencies. The LSTM processes the frame sequence in one direction, while the BiLSTM processes the sequence in both directions [19]. Stacked LSTMs are used in this study to enhance the model's ability to learn complex relationships in sequential data. In these layers, the output shape becomes (75, X), where 75 is the number of timesteps and X is the number of units in LSTM/BiLSTM which is determined through hyperparameter tuning.

Next, a TimeDistributed Dense layer with 128 units and a ReLU activation function was applied, producing an output of shape (75, 128), where 75 represents the timesteps and 128 represents the number of units in Dense layer. Lastly, a TimeDistributed Dense layer with a Softmax activation function, which is responsible to generate frame-level predictions as probability distributions for every vocabulary and blank token, was applied. It results in an output shape of (75, 45), where 75 represents the timesteps and 45 represents the number of vocabulary in the dataset. During the training, CTC loss function was applied to handle the mismatch between input frames and predicted word lengths. Finally, a decoding process was done using the argmax function to convert the probability outputs into a sequence of predicted words.

After defining the model architecture, hyperparameter tuning was conducted to obtain the most optimal hyperparameter configurations for the 3DCNN-LSTM and 3DCNN-BiLSTM model. This process ensure that the models were trained using the best combination of hyperparameter, so that the models are able to recognize sentences in BISINDO with the lowest possible Word Error Rate (WER). In this study, Bayesian Optimization (BO) was used for hyperparameter tuning due to its efficiency in finding optimal configurations compared to methods like Random Search [20] or Grid Search, while maintaining equal or better model accuracy [21]. The hyperparameters tuned in this research includes the number of LSTM/BiLSTM units, learning rate, batch size, and dropout rate. The predefined ranges for these hyperparameters are shown in Table V.

TABLE V. OPTIMIZED HYPERPARAMETERS

Hyperparameter	Value
LSTM/BiLSTM units	64, 128, 256 units
Learning Rate	0.001, 0.0005, 0.0001
Batch Size	16, 32, 64
Dropout Rate	0.2, 0.3, 0.4

Once the optimal combination of hyperparameter was found for both the 3DCNN-LSTM and 3DCNN-BiLSTM models, the final models were built using these values and training was initiated. This ensured the models were trained under the best configuration, which allows the model to maximize their ability in recognizing word sequences in BISINDO.

G. Evaluation

To evaluate the model against the given input sentences, the metric used in this research is Word Error Rate (WER). This metric is a standard measure commonly used to evaluate the performance of a machine translation system or speech recognition system.

The WER measurement helps evaluate the difference between the output sequence (the identified sentences) and the actual sentences by taking into account substitutions (instances where identified word is different from the actual word intended to be conveyed through sign gestures), deletions (instances where a signed word is missing from the identified output sequence), and insertions (instances where an extra word appears in the output sequence that was not present in the signed gesture). The formula for measuring WER is as follows:

$$WER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C}$$

where,

- S represents the number of substitutions
- D represents the number of deletions
- I represents the number of insertions
- C represents the number of correctly identified words

By using WER as an evaluation metric, researchers can determine how many errors or mistakes the system makes in identifying sentences in sign language, allowing for a more accurate assessment of the model's performance. A smaller WER value indicates fewer errors in the identified output, meaning that a lower WER reflects better system performance.

IV. RESULTS

This chapter presents the results of the sentence-level SLR model, which was trained and evaluated using the WER metric. Before training the model, hyperparameter tuning was done using Bayesian Optimization to find the best hyperparameter combination for both the 3DCNN-LSTM and 3DCNN-BiLSTM models. The tuning process was carried out with 15 trials for each model. The results of the hyperparameter tuning trials for the LSTM model can be seen in Table VI, and the results for the BiLSTM model can be seen in Table VII.

TABLE VI. LSTM MODEL HYPERPARAMETER TUNING TRIALS

Trial ID	LSTM Units	Dropout	Batch Size	Learning Rate	Score
0	256	0.3	32	0.0005	11.93222332
1	64	0.3	64	0.0005	14.42325211
2	64	0.2	16	0.0005	12.05967999
3	256	0.2	16	0.0005	12.19359112
4	256	0.2	64	0.001	12.75675106
5	128	0.4	16	0.0005	12.39139652
6	128	0.3	16	0.0005	11.87120438
7	256	0.2	64	0.0001	16.38278389
8	128	0.2	64	0.0001	16.72937202
9	128	0.4	64	0.0005	13.32501125
10	256	0.4	64	0.0005	13.03599548
11	128	0.4	64	0.0001	16.78538322
12	256	0.4	32	0.001	12.30996895
13	**128**	**0.2**	**16**	**0.001**	**11.21184921**
14	64	0.3	16	0.001	11.78033161

TABLE VII. BILSTM MODEL HYPERPARAMETER TUNING TRIALS

Trial ID	LSTM Units	Dropout	Batch Size	Learning Rate	Score
0	64	0.3	32	0.0001	12.86169529
1	128	0.4	16	0.0005	9.067760468
2	64	0.4	64	0.0005	11.28538609
3	256	0.4	64	0.001	8.874022484
4	256	0.4	32	0.0001	9.88320446
5	128	0.2	16	0.0005	9.092283249
6	256	0.3	16	0.0005	8.945654869
7	256	0.4	64	0.0005	9.24458313
8	128	0.3	64	0.001	9.18018055
9	256	0.3	64	0.0005	9.229111671
10	256	0.2	64	0.0001	10.96395779
11	128	0.2	32	0.001	9.443668365
12	**256**	**0.4**	**16**	**0.001**	**8.634145737**
13	256	0.2	16	0.001	8.818277359
14	64	0.3	16	0.001	9.163191795

Tables VI and VII show the results of hyperparameter tuning for the LSTM and BiLSTM models, where each trial tests a different combination of LSTM units, dropout rate, batch size, and learning rate. The "Score" column represents the validation loss, meaning that lower values indicate better model performance on unseen data. Based on Tables VI and VII, it can be seen that the best hyperparameter combination for the LSTM model consists of 128 LSTM units, a dropout rate of 0.2, a batch size of 16, and a learning rate of 0.001. Meanwhile, the BiLSTM model achieved its best performance with 256 LSTM units, a dropout rate of 0.4, a batch size of 16, and a learning rate of 0.001. These hyperparameter

combinations were then used to build the models for training. Accordingly, the final structure of the 3DCNN-LSTM model can be seen in Fig. 4, and the final structure of the 3DCNN-BiLSTM model can be seen in Fig. 5.

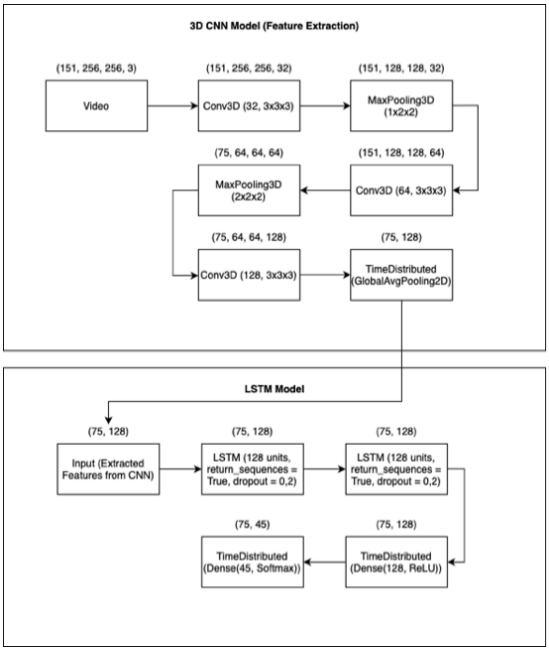


Fig. 4. Hybrid 3DCNN-LSTM model structure.

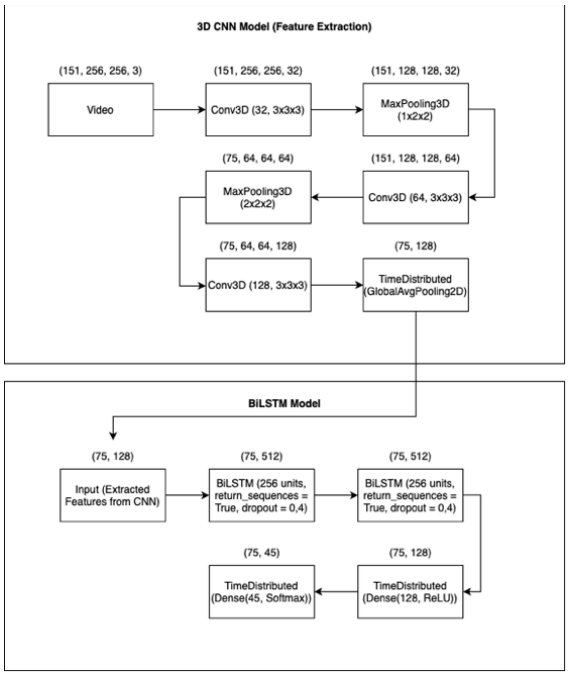


Fig. 5. Hybrid 3DCNN-BiLSTM model structure.

Each model was trained for 200 epochs. To prevent overfitting, early stopping with a patience of 20 was applied by monitoring the validation loss. The LSTM model achieved its lowest validation loss at epoch 187, with a value of 5.7631. The training and validation loss of the LSTM model can be seen in Fig. 6.

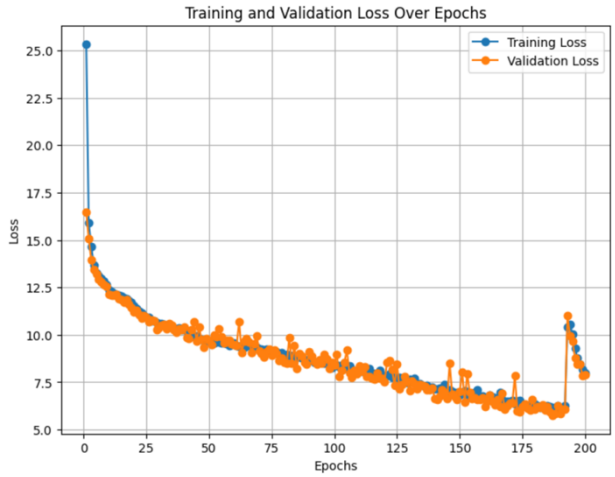


Fig. 6. Training and validation loss of LSTM model.

The graph shows that both training and validation loss decreased steadily, indicating that the model was learning well. However, at epoch 193, there was a sudden spike in both losses, which later dropped again. This spike was likely caused by temporary fluctuations. Since the losses continued to decline afterwards, the model was still learning effectively. In this case, the best weights were used, taken from before the spike occurred.

Meanwhile, the BiLSTM model achieved its lowest validation loss at epoch 183, with a value of 0.2444. The training and validation loss of the BiLSTM model can be seen in Fig. 7.

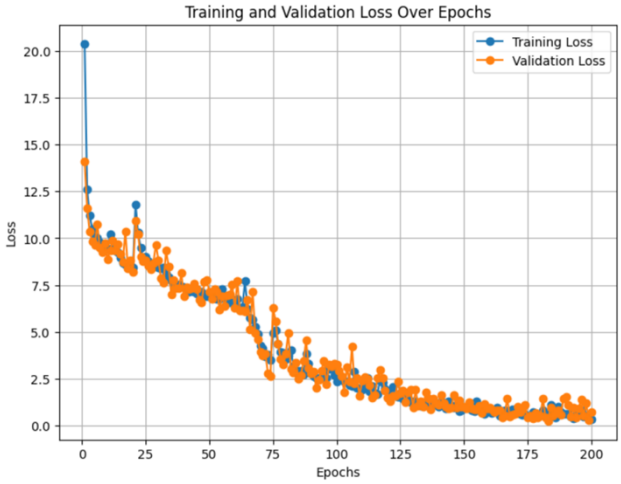


Fig. 7. Training and validation loss of BiLSTM model.

The graph shows a consistent decrease in both training and validation loss, indicating that the model was learning well from the data. Although some fluctuations occurred during training, overall the loss decreased steadily, with the training and validation loss values remaining close throughout the process.

After training and selecting the best weights for both models, we evaluated them on the test set to measure their performance. The predictions from this testing phase were

decoded to convert the model's output into sentence form. Once the predicted word sequences were obtained, the model's performance was evaluated using the WER metric, which is calculated based on the number of prediction errors, including substitutions, insertions, and deletions.

In the 3DCNN-LSTM model, the evaluation on the testing set resulted in a relatively high WER of 59.21%. It produced the most errors in substitutions with a total of 211 substitutions, followed by 60 deletions and 28 insertion errors. The error distribution for the LSTM model can be seen in Fig. 8.

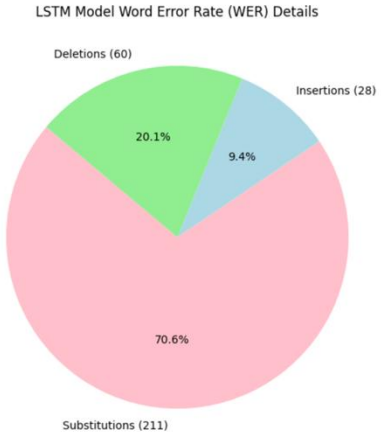


Fig. 8. Error percentage of the LSTM model on the test set.

Meanwhile, the 3DCNN-BiLSTM model achieved a better performance with a significantly lower WER of 2.77%, with the most errors in substitution with a total of 12 substitutions, followed by two insertion errors. The 3DCNN-BiLSTM model did not produce any deletion errors. The error distribution for the BiLSTM model can be seen in Fig. 9.

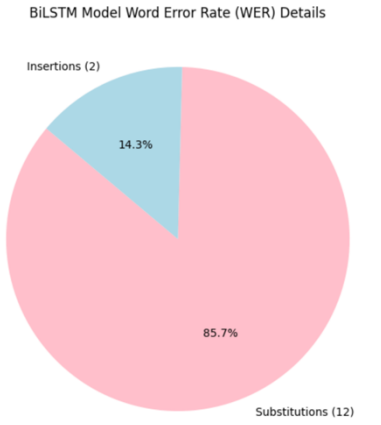


Fig. 9. Error percentage of the BiLSTM model on the test set.

To address the limitation of having only one signer in the dataset, additional signer videos with plain backgrounds and clear visuals were collected from publicly available sources on the internet to serve as a supplementary test set. We successfully collected seven videos of BISINDO DKI Jakarta sentences to be used for the additional test.

In testing using the additional signer videos collected from the internet, the LSTM model achieved a WER of 120% with 11 substitution errors, followed by 5 insertions and 2 deletions errors. Meanwhile, the BiLSTM model achieved a WER of 60%, with 9 substitution errors without insertion and deletion errors.

V. Discussion

Both 3D CNN-LSTM and 3D CNN-BiLSTM models were trained using the hyperparameters determined from tuning, ensuring that each model was optimized for its architecture. Both models were trained for the same number of epochs, which is 200 epochs, to allow a fair comparison. The average training time per epoch was 8.66 seconds for the LSTM and 11 seconds for BiLSTM model. The BiLSTM model contains a larger number of trainable parameters than the LSTM, which contributes to the slightly longer training time. While BiLSTM requires slightly more time due to its bidirectional architecture, the difference is small and does not significantly affect the fairness of the comparison.

Based on the evaluation results of the LSTM and BiLSTM models, there is a significant difference in WER between the two models. The 3D CNN-BiLSTM model shows a drastically lower WER compared to the 3D CNN-LSTM model. As seen from the training loss, the LSTM loss was still decreasing at the end of 200 epochs, indicating that it could potentially achieve slightly better performance if trained for more epochs. However, the BiLSTM still outperforms the LSTM due to its ability to capture context in both directions.

In sentence-level SLR, the meaning of a gesture often depends on both what comes before and after, creating context across the whole sentence. While the LSTM can only utilize past frames, the BiLSTM may look at the sequence in both directions. This bidirectional processing helps BiLSTM to distinguish gestures that might look similar on their own and better capture subtle differences in hand movements and facial expressions. Even if the LSTM is trained longer, LSTM is still limited in using future context, which explains the drastic performance difference between the models. This also shows that bidirectional modeling is so important for sentence-level SLR.

However, the significantly low WER on the BiLSTM model may also be due to the lack of signer variation in the dataset, meaning the model has effectively learned the patterns of a single individual, which allows it to predict word sequences in the test set very well. This becomes clear when testing on videos from new signers, where the WER is higher than on the original test set. However, this should not be considered as the main benchmark, as uncontrolled factors in videos collected from social media can affect the model's ability to recognize sign language. The differences in signing style may also contribute to the model's difficulty in accurately recognizing gestures in videos from new signers. Although the model cannot yet fully recognize the word sequences in sign language sentences, it can be seen that it can fairly well recognize and identify sentence length. This is demonstrated by the small number of insertion and deletion errors made by both models, while both models make more substitution errors. This demonstrates that overall sentence length can be identified well, although errors in sign recognition still occur, which can be due to various factors, such as similar gestures.

Nevertheless, compared to previous similar work, such as sentence-level SLR using 3D CNN-BiRNN model, the 3D CNN-BiLSTM model still shows a lower WER, even when testing on videos from different signers. This improvement is likely due to the LSTM's gating mechanisms, which better handle vanishing gradients, allowing the model to remember important information over longer sequences better than BiRNN.

VI. Conclusion

This study presents a novel approach to sentence-level BISINDO DKI Jakarta sign language recognition by combining a 3D CNN model for spatial and temporal feature extraction, followed by a LSTM/BiLSTM network to capture sequential dependencies, enhanced by CTC, which allows for sentence-level sign language recognition without the need for word-level annotations. Both 3DCNN-LSTM and 3DCNN-BiLSTM model was trained and evaluated on the 30 BISINDO sentence videos consisting of everyday phrases, with data augmentation applied to improve generalization and hyperparameters tuned to optimize performance.

The research was conducted through several stages. First, the feature extraction was performed by the 3D CNN model, with the resulting features stored in a .npy file. These features are then passed to two different models – LSTM and BiLSTM – for training and testing in sentence-level BISINDO recognition. Both models were built using optimal hyperparameter configurations obtained through Bayesian Optimization. From the experiments, several key findings can be drawn:

1) The 3DCNN-LSTM model was not able to achieve a good recognition performance, indicated by a relatively high WER of 59.21% on the testing set. When evaluated on new signer videos, the model produced an even higher WER of 120%, demonstrating that it was unable to effectively recognize sign language, particularly when faced with data from different signers.

2) The 3DCNN-BiLSTM model showed significantly better recognition performance than the 3DCNN-LSTM. It achieved a low WER of 2.77% on the testing set, indicating strong performance for recognizing sentences from a single signer. However, when tested on new signer videos, it achieved a higher WER of 60%. While it is still not highly accurate in recognizing unseen signers, the BiLSTM outperformed the LSTM. This is likely due to BiLSTM's ability to capture contextual information in both forward and backward directions.

3) The combination of 3D CNN as a feature extractor and BiLSTM as a sequence model demonstrated promising results for sentence-level BISINDO recognition in the testing set. However, this outstanding performance is still influenced by the limitations of the dataset, especially the limited number of signers, which causes the model to overfit to specific movement patterns.

4) Although neither the 3DCNN-LSTM nor the 3DCNN-BiLSTM achieved perfect recognition of BISINDO sentences, both models were relatively effective in capturing sentence length. This was reflected by fewer insertion and deletion errors compared to substitution errors.

One of the main limitations in this study is the very limited dataset size, which only involves one non-native signer, so the model's ability to generalize to various variations of signer gestures when demonstrating sign language cannot be fully proven. Therefore, in the future, it is recommended for researchers to develop the dataset by adding more signers so that the model can learn the variations of gestures of several signers who have their own unique characteristics. Furthermore, the number of sentence samples should be increased to allow the model to learn from a broader variety of expressions. By expanding the dataset, the resulting model will be able to recognize a wider range of sign language gestures.

Furthermore, based on the excellent results of the 3DCNN-BiLSTM model evaluation in recognizing sign language sentences on a single signer test set, the development of personalized sign language recognition can also be considered in future research. As the number of datasets and the variety of signers increase, researchers can also develop this recognition model into a system with a broader scope for public use. For example, this research can be further developed into a web-based or mobile application that can be used as a medium for learning sign language in the community, to become a reliable communication medium for the deaf and hearing communities.

Acknowledgment

The authors would like to thank Bina Nusantara University, Jakarta, for their support in providing the research facilities.

Authors' Contribution

Katriel Larissa Wiguna: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Visualization. Rojali: Writing – Review & Editing, Supervision.

Data Availability

Data supporting this study are not publicly available because they contain identifiable facial information, which raises privacy concerns.

References

[1] A. S. Nugraheni, A. P. Husain, and H. Unayah, "Optimalisasi Penggunaan Bahasa Isyarat Dengan SIBI DAN BISINDO Pada Mahasiswa Difabel Tunarungu Di Prodi PGMI UIN Sunan Kalijaga," Jurnal Holistika, vol. 5, no. 1, p. 28, Feb. 2023, doi: 10.24853/holistika.5.1.28-33.

[2] A. Wadhawan and P. Kumar, "Sign Language Recognition Systems: A Decade Systematic Literature Review," Archives of Computational Methods in Engineering, vol. 28, no. 3, pp. 785–813, May 2021, doi: 10.1007/s11831-019-09384-2.

[3] R. S. L. Murali, L. D. Ramayya, and V. A. Santosh, "Sign Language Recognition System Using Convolutional Neural Network And Computer Vision," International Journal of Engineering Innovations in Advanced Technology, vol. 4, no. 4, pp. 137–142, Dec. 2022.

[4] M. C. Ariesta, F. Wiryana, Suharjito, and A. Zahra, "Sentence Level Indonesian Sign Language Recognition Using 3D Convolutional Neural Network and Bidirectional Recurrent Neural Network," in 2018 Indonesian Association for Pattern Recognition International

Conference (INAPR), IEEE, Sep. 2018, pp. 16–22. doi: 10.1109/INAPR.2018.8627016.

[5] J. Shin et al., "Korean Sign Language Recognition Using Transformer-Based Deep Neural Network," Applied Sciences, vol. 13, no. 5, p. 3029, Feb. 2023, doi: 10.3390/app13053029.

[6] P. Vyavahare, S. Dhawale, P. Takale, V. Koli, B. Kanawade, and S. Khonde, "Detection and Interpretation of Indian Sign Language Using LSTM Networks," Journal of Intelligent Systems and Control, vol. 2, no. 3, pp. 132–142, Jul. 2023, doi: 10.56578/jisc020302.

[7] G. Latif, N. Mohammad, R. AlKhalaf, R. AlKhalaf, J. Alghazo, and M. Khan, "An Automatic Arabic Sign Language Recognition System based on Deep CNN: An Assistive System for the Deaf and Hard of Hearing," International Journal of Computing and Digital Systems, vol. 9, no. 4, pp. 715–724, Jul. 2020, doi: 10.12785/ijcds/090418.

[8] Q. Xiao, X. Chang, X. Zhang, and X. Liu, "Multi-Information Spatial–Temporal LSTM Fusion Continuous Sign Language Neural Machine Translation," IEEE Access, vol. 8, pp. 216718–216728, 2020, doi: 10.1109/ACCESS.2020.3039539.

[9] G. S. Sajja, S. R. Addula, M. K. Meesala, and P. Ravipati, "Optimizing inventory management through AI-driven demand forecasting for improved supply chain responsiveness and accuracy," 2025, p. 050003. doi: 10.1063/5.0275697.

[10] Badan Pusat Statistik, "Kepadatan Penduduk menurut Provinsi (jiwa/km2), 2021."

[11] A. Aljabar and S. Suharjito, "BISINDO (Bahasa Isyarat Indonesia) Sign Language Recognition Using CNN and LSTM," Advances in Science, Technology and Engineering Systems Journal, vol. 5, no. 5, pp. 282–287, 2020, doi: 10.25046/aj050535.

[12] R. A. Alawwad, O. Bchir, and M. Maher, "Arabic Sign Language Recognition using Faster R-CNN," International Journal of Advanced Computer Science and Applications, vol. 12, no. 3, 2021, doi: 10.14569/IJACSA.2021.0120380.

[13] L. Arisandi and B. Satya, "Sistem Klarifikasi Bahasa Isyarat Indonesia (Bisindo) Dengan Menggunakan Algoritma Convolutional Neural Network," Jurnal Sistem Cerdas, vol. 5, no. 3, pp. 135–146, Dec. 2022, doi: 10.37396/jsc.v5i3.262.

[14] D. Kothadiya, C. Bhatt, K. Sapariya, K. Patel, A.-B. Gil-González, and J. M. Corchado, "Deepsign: Sign Language Detection and Recognition Using Deep Learning," Electronics (Basel), vol. 11, no. 11, p. 1780, Jun. 2022, doi: 10.3390/electronics11111780.

[15] T. H. Noor et al., "Real-Time Arabic Sign Language Recognition Using a Hybrid Deep Learning Model," Sensors, vol. 24, no. 11, p. 3683, Jun. 2024, doi: 10.3390/s24113683.

[16] E. N. Siagian, "Kata Berfrekuensi Tinggi dalam Pembelajaran BIPA Pemula," Ranah: Jurnal Kajian Bahasa, vol. 9, no. 2, p. 188, Dec. 2020, doi: 10.26499/rnh.v9i2.2320.

[17] A. M. Abdulghani, M. M. Abdulghani, W. L. Walters, and K. H. Abed, "Data Augmentation Using Brightness and Darkness to Enhance the Performance of YOLO7 Object Detection Algorithm," in 2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE), IEEE, Jul. 2023, pp. 351–356. doi: 10.1109/CSCE60160.2023.00061.

[18] N. B. Ibrahim, H. H. Zayed, and M. M. Selim, "Advances, Challenges and Opportunities in Continuous Sign Language Recognition," Journal of Engineering and Applied Sciences, vol. 15, no. 5, pp. 1205–1227, Dec. 2019, doi: 10.36478/jeasci.2020.1205.1227.

[19] R. L. Abduljabbar, H. Dia, and P.-W. Tsai, "Unidirectional and Bidirectional LSTM Models for Short-Term Traffic Prediction," J Adv Transp, vol. 2021, pp. 1–16, Mar. 2021, doi: 10.1155/2021/5589075.

[20] R. Turner et al., "Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020," Apr. 2021.

[21] A. Stuke, P. Rinke, and M. Todorović, "Efficient hyperparameter tuning for kernel ridge regression with Bayesian optimization," Apr. 2020.

Formal Verification of a Blockchain-Based Security Model for Personal Data Sharing Using the Dolev-Yao Model and ProVerif

Godwin Mandinyenya, Vusumuzi Malele
School of Computer Science and Information Systems, North-West University, Vanderbijlpark, South Africa

***Abstract*—Secure personal data sharing remains a critical challenge in decentralized systems due to concerns over privacy, compliance, and trust. This paper presents the formal verification of a Blockchain-Based Security Model (BSM) designed to address these challenges through a multi-layered architecture. The proposed model integrates Chaincode-as-a-Service (CCaaS) on Hyperledger Fabric to ensure modular, maintainable, and scalable execution of smart contracts. A Flask-based API serves as the secure gateway for data operations and identity management. Sensitive data is stored off-chain using InterPlanetary File System (IPFS), preserving decentralization while minimizing on-chain bloat. Access control is enforced using efficient cryptographic techniques, while Intel SGX (or simulated enclaves) safeguards secure data processing and decryption within trusted execution environments. To further enhance privacy guarantees, Zero-Knowledge Proofs (ZKPs) are optionally integrated to enable verifiable claims without disclosing raw data. For assurance of correctness and security, the BSM is formally modeled using the Dolev-Yao attacker model and verified through ProVerif, focusing on key security properties such as confidentiality, integrity, authentication, and accountability. The findings confirm that the proposed model satisfies stringent security goals and is robust against symbolic adversaries. This work contributes a verifiable and extensible framework for privacy-preserving data sharing in sectors such as healthcare, finance, and government. To the best of our knowledge, this is among the first works to formally verify a blockchain-based security model that simultaneously integrates modular chaincode execution (CCaaS), trusted hardware enclaves (Intel SGX), decentralized off-chain storage (IPFS), and optional Zero-Knowledge Proofs (ZKPs) with a unified framework for personal data sharing.**

Keywords*—Blockchain; security model; Chaincode-as-a-Service; InterPlanetary File System; Intel Software Guard Extensions; Zero-Knowledge Proofs ProVerif; formal verification; Dolev-Yao

I. Introduction

In the digital era, the exponential growth in data generation has led to a parallel rise in privacy concerns, especially in domains involving personal information such as healthcare, education, finance, and identity management. Individuals, institutions, and governments are increasingly reliant on digital platforms for the storage, processing, and sharing of sensitive personal data. However, traditional centralized architectures used to manage these transactions are plagued by significant security vulnerabilities, ranging from unauthorized access and data breaches to single points of failure and non-transparent access control mechanisms. In this context, blockchain technology has emerged as a transformative solution capable of decentralizing trust and enhancing data integrity, accountability, and user autonomy [1].

Blockchain-based systems, particularly those built on platforms like Hyperledger Fabric, offer programmable capabilities through smart contracts, specifically Chaincode-as-a-Service (CCaaS). These smart contracts facilitate tamper-proof transaction logic and offer fine-grained control over data access and updates in distributed environments [2]. While public blockchains like Ethereum focus on openness and censorship resistance, private and permissioned blockchains like Hyperledger Fabric prioritize scalability, enterprise-grade access control, and modular architecture, making them more suitable for secure personal data sharing scenarios [3].

Despite these advantages, current blockchain implementations are often limited in their ability to balance privacy, scalability, and compliance with data protection regulations such as the General Data Protection Regulation (GDPR). To address this, researchers have proposed hybrid architectures that combine on-chain verification with off-chain storage using tools like the InterPlanetary File Systems (IPFS) [4]. IPFS reduces blockchain bloat while enabling cryptographically verifiable file storage, offering a lightweight method to decentralize large personal datasets while maintaining their integrity.

In addition, cryptographic control mechanisms such as attribute based encryption and Zero-Knowledge Proofs (ZKPs) have been explored to enforce fine-grained data access without revealing sensitive attributes [5]. Meanwhile, Intel Software Guard Extensions (SGX) provides a secure hardware-based enclave for confidential computation, further strengthening end-to-end data protection [6]. Together, these technologies can form a powerful privacy-preserving architecture.

Formal verification becomes critical in this context. Unlike conventional testing, which checks for specific failures, formal methods methematically prove whether a system satisfies certain security properties under well-defined adversarial models. Among the most widely accepted frameworks for such verification in cryptographic protocol analysis is the Dolev-Yao model, which assumes the attacker has full control of the network but cannot break cryptographic primitives [7]. Coupled with tools like ProVerif, this model allows the symbolic analysis of authentication, confidentiality, integrity, and other critical properties in complex protocols [8].

This paper presents a formally verified blockchain-based security model for personal data sharing, developed with Chaincode-as-a-Serive on Hyperledger Fabric, integrated with IPFS for off-chain storage, cryptographic access control policies, and trusted enclave-based computation via Intel SGX. We explore how the formal application of the Dolev-Yao model using ProVerif validates the mode's resilience to classical adversarial threats such as man-in-the-middle attacks, replay attacks, and data leakage through side channels. The model also includes optional integration of ZKPs to extend verifiability in cases of sensitive identity disclosure or regulatory audit requirements.

The motivation for this research is threefold. First, there is a significant gap in formally verified blockchain architectures that support composable and modular integration of cryptographic enforcement techniques for personal data [9]. Second, existing solutions lack robust verification of hardware-backed secure enclaves within hybrid architectures. While SGX provides protection at the hardware level, it is imperative that these components are also modeled symbolically to validate system-level properties [10]. Third, cross-jurisdictional regulations demand adaptive and transparent systems [11]. Secure personal data sharing remains a pressing challenge, requiring unified architectures that balance privacy, scalability, and verifiable security.

The present work addresses this gap by designing and verifying a Blockchain-Based Security Model (BSM) that integrates Chaincode-as-a-Service (CCaaS), Intel SGX enclaves, InterPlanetary File System (IPFS) storage, and optional Zero-Knowledge Proofs (ZKPs). The design is validated using the Dolev–Yao model and the ProVerif tool, enabling mathematical proofs of confidentiality, integrity, authentication, authorization, and auditability.

In summary, this paper presents the design and formal verification of a Blockchain Security Model that integrates CCaaS, SGX, IPFS, and optional ZKPs. The remainder of the paper is organized as follows: Section II reviews related work; Section III presents the methodology; Section IV details the model and verification approach; Section V reports results; Section VI discusses implications; and Section VII concludes the paper.

The study is guided by the following research questions:

- RQ1: How can CCaaS support transparent and modular enforcement of access controls in personal data sharing?
- RQ2: How do IPFS and Intel SGX improve the scalability and confidentiality of the security model?
- RQ3: Can ZKPs enhance privacy without degrading system performance?
- RQ4: To what extent does ProVerif verify key security properties of the BSM under the Dolev-Yao model?
- RQ5: What trade-offs exist between security, performance, and regulatory compliance in the proposed model?

The main contributions of this paper are as follows:

1) We design a modular Blockchain Security Model (BSM) integrating CCaaS, IPFS, and Intel SGX, with optional ZKPs for privacy-preserving verification.

2) We formally verify the model under the Dolev-Yao adversarial model using ProVerif, demonstrating confidentiality, integrity, authentication, and accountability.

3) We evaluate practical performance trade-offs across CCaaS, SGX, IPFS, and ZKP modules, confirming the model's viability in privacy-sensitive domains such as healthcare, finance and e-governement.

A. Research Problem

Despite advancements in blockchain architectures, existing models lack formal verification and integrated privacy-preserving mechanisms combining CCaaS, SGX, IPFS, and ZKPs. This leaves gaps in trust, compliance, and deployability across sensitive domains.

B. Research Objectives

The study was guided by the following research objective:

1) To design a modular blockchain security model integrating CCaaS, SGX, IPFS, and ZKPs.

2) To formally verify its security properties under the Dolev-Yao model using ProVerif.

3) To evaluate its performance and compliance in practical scenarios.

C. Significance

This study provides one of the first formally verified frameworks for privacy-preserving blockchain-based data sharing. It contributes a deployable architecture that can be trusted by organizations and regulators alike.

II. Related Work

In recent years, extensive research has been dedicated to enhancing privacy and security in blockchain-based personal data sharing systems. These studies span multiple dimensions, including on-chain governance, access control, secure enclaves, and formal verification techniques. However, few have proposed integrated, end-to-end solutions that combine robust cryptographic techniques with formal analysis using Dolev-Yao model and ProVerif.

Several blockchain solutions have emerged focusing on data privacy and decentralized identity. For example, Belchior et al. [12] surveyed interoperability efforts in blockchain identity systems, highlighting significant gaps in secure personal data exchange, especially under dynamic policy constraints. Similarly, Zwitter and Boisse-Despiaux [13] emphasized the importance of transparency and accountability mechanisms for data management in decentralized platforms, aligning with the GDPR's principles of lawful processing.

To enforce fine-grained access controls, cryptographic primitives like Attribute-Based Encryption (ABE) and Proxy Re-Encryption (PRE) have been employed in multiple works

[14], [15]. However, these models often lack verifiability of enforcement and suffer from poor scalability. Recent frameworks have begun to integrate decentralized storage, such as IPFS, to mitigate blockchain storage limitations [16]. Yet, challenges persist in ensuring secure off-chain computation and auditing.

Intel SGX has been widely adopted to secure data processing via trusted execution environments (TEEs), particularly in scenarios requiring computation on encrypted data [17]. Projects like Ekiden [18] and Oasis Labs [19] exemplify the utility of SGX in enabling privacy-preserving smart contracts. However, these models either remain proprietary or insufficiently validated under formal adversarial models.

Zero-Knowledge Proofs (ZKPs) have also gained prominence as privacy-preserving tools in blockchain applications. Systems like Zcash and zkSync demonstrate their potential in hiding sensitive attributes during transactions [20]. Yet, these systems focus on financial use cases and do not generalize well to the broader context of personal data sharing. Furthermore, the integration of ZKPs with access control and accountability layers remains underexplored.

Regarding smart contract modularization, Chaincode-as-a-Service (CCaaS) has recently been proposed in Hyperledger Fabric to separate application logic from blockchain nodes [21]. While CCaaS offers architectural flexibility, little work has been done to assess its security implications in multi-tenant environments or its resilience to message tampering under adversarial conditions.

Formal verification of blockchain protocols has become increasingly vital for ensuring provable security. Tools such as ProVerif and Tamarin have been employed to validate consensus algorithms, voting protocols, and authentication schemes [22], [23]. Nevertheless, comprehensive verification of integrated blockchain models, incorporating chaincode, SGX, IPFS, and ZKPs, remains largely uncharted.

Recent studies continue to highlight challenges in applying symbolic verification to blockchain-based architectures. For example, Zhang et al. [34] analysed compositional verification challenges in multi-chain environments, while Liu et al. [36] demonstrated the difficulty of modelling enclave-based side-channel attacks within the Dolev-Yao abstraction. Similarly, Wood et al. [33] emphasized the limitations of zk-SNARK integration in blockchain systems under symbolic analysis. These challenges underscore that while the Dolev-Yao model provides conservative guarantees, its abstraction excludes low-level hardware exploits and performance-related trade-offs, which remain open problems for future research.

In light of these limitations, our research offers a holistic security model that not only combines modular chaincode execution (via CCaaS), decentralized off-chain storage (via IPFS), secure enclaves (Intel SGX), and optional ZKPs, but also conducts formal verification using the Dolev-Yao threat model implemented in ProVerif. Unlike most existing systems, we explicitly model and verify access control correctness, data confidentiality, and policy compliance under active adversarial conditions. This work advances the state-of-the-art by providing both theoretical gurantees and practical deployability within regulated environments, bridging the gap between academic models and production-grade systems.

III. Methodology

This study employs a Design Science Research (DSR) methodology to systematically design, implement, and formally verify a blockchain-based security model (BSM) tailored for secure personal data sharing. The methodology comprises five interlinked phases: problem identification, artifact design, development, validation, and contribution analysis. Each phase is structured to ensure scientific rigor, technical feasibility, and alignment with regulatory and privacy mandates such as GDPR and the African Union Convention on Cyber Security and Personal Data Protection. The research methodology adopted in this study follows the Design Science Research (DSR) paradigm, as illustrated in Fig. 1.

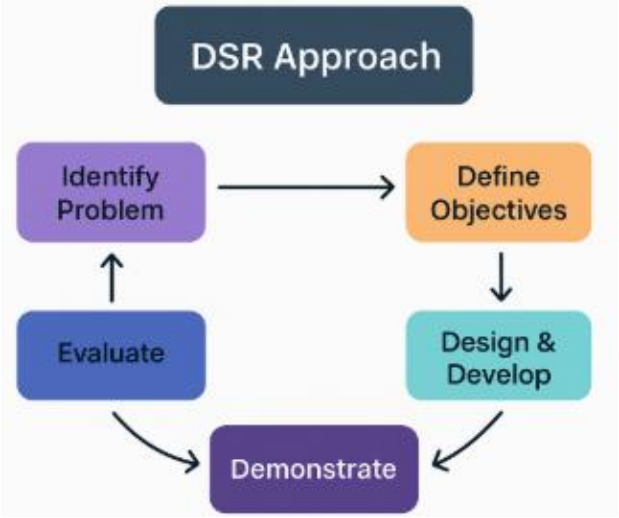


Fig. 1. Design Science Research (DSR) methodology followed in this study. (adapted from [12]).

A. Research Framework

The DSR framework is selected for its iterative construct-evaluate-refine approach suitable for engineering artifacts that bridge theory and practice [24]. The formal modeling component, critical to this study, is guided by the Dolev-Yao attacker model, a symbolic abstraction widely adopted in formal security proofs [25], and evaluated using ProVerif, a state-of-the-art automated cryptographic protocol verifier [26].

IV. Model Design and Development

A. Blockchain Infrastructure

The system is implemented using Hyperledger Fabric v2.5, with a modular CCaaS deployment allowing smart contracts to be hosted and invoked dynamically via RESTful Flask-based APIs. This architecture promotes maintainability, service abstraction, and network governance separation, critical for permissioned consortia networks [27]. The `grantAccess()` function below demonstrates how Chaincode-as-a-Service (CCasS) enforces attribute-based access control by checking if the transaction invoker holds the `admin` role before writing access permissions to the ledger.

The access control logic is implemented in CCaaS chaincode. The GrantAccess procedure, shown in Algorithm 1, ensures that only administrators can authorize access to a specific content identifier (CICD).

Algorithm 1: Sample CCaaS Chaincode Grant Function in Go

```
func (s *SmartContract) GrantAccess(
contractapi.TransactionContextInterface,
userID string, cid string) error {
    // Check if the invoker has the 'admin' attribute
    attrValue, found, err :=
            if !found || attrValue != "admin" {
            return fmt.Errorf("Failed to get
                attribute 'role': %v", err)
                if !found || attrValue != "admin" {
                return fmt.Errorf("Only users with
                admin role can grant access")}
                // Construct access key based on userID
                accessKey :=  fmt.Sprintf("access_%s_%s",
                userID, cid)
                // Store access flag
                err = ctx.GetStub().PutState(accessKey
                []byte("granted"))
                if err != nil {
                return fmt.Errorf("Failed to grant
                access: %v", err)}
                return nil
            End
    End
```

In practice, the GrantAccess () function enforces consent-driven access policies. For example, in a healthcare scenario, only users with the admin attribute (e.g. hospital administrators) can authorise doctors to retrieve patient records from IPFS. This logic is aligned with Hyperledger Fabric's MSP-based role enforcement [2] and ensures that access decisions are both transparent and auditable on-chain. By embedding this algorithm within CCaaS, the model operationalizes GDPR-aligned consent management at the chaincode level.

B. Off-Chain Storage via IPFS

To address scalability and privacy challenges, sensitive data is encrypted and stored off-chain using the InterPlanetary File System (IPFS) [30]. Only metadata, content identifiers (CIDs), and smart contract state changes are committed to the blockchain, achieving a verifiable audit trail without overburdening the ledger [28].

C. Access Control via Cryptography

Access control is enforced through hybrid Attribute-Based Encryption (ABE) and public-key infrastructure (PKI) techniques. Policy metadata is embedded in chaincode logic, and decryption keys are issued via authorized Certificate Authorities (CAs) based on user roles and data access permissions [29].

D. Secure Computation with Intel SGX

The design integrates Intel SGX enclaves (simulated for current testing) to process sensitive data and decryption requests in a hardware-isolated environment. This ensures that even with system compromise, decrypted data and keys remain confidential and auditable [30].

E. Optional Zero-Knowledge Proofs (ZKPs)

To enhance privacy-preserving verifiability, ZKPs are optionally embedded to prove compliance with access conditions without revealing user attributes or transaction content. The ZKP layer uses zk-SNARKs, simulated using ZoKrates to verify logic without exposing data [31]. The architecture consists of four core layers: (i) User layer, handling authentication and data submission; (ii) Application Layer, implemented via Flask APIs that interface with chaincode and SGX enclaves; (iii) Blockchain Layer, where Hyperledger Fabric maintains immutable logs and CCaaS executes business logic; and (iv) Storage and Verification Layer, composed of IPFS for off-chain encrypted storage and ProVerif for formal verification under the Dolev-Yao model. An optional ZKP module enables privacy-preserving access validation. The system architecture is shown in Fig. 2.

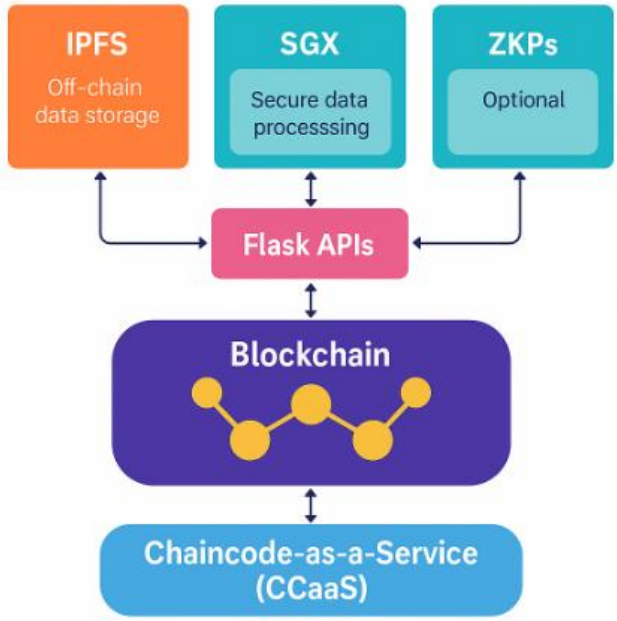


Fig. 2. System architecture of the blockchain-based security model integrating Chaincode-as-a-Service (CCaaS), IPFS, SGX, Flask APIs, and optional Zero-Knowledge Proofs (ZKPs).

F. Formal Verification Using Dolev-Yao and ProVerif

1) The formal model captures entities (users, CA, SGX, IPFS, peers), messages, and cryptographic operations using applied pi-calculus, the input language for ProVerif. The symbolic model assumes a powerful adversary per Dolev–Yao capabilities, able to intercept, modify, and forge messages over the network [25].

2) Security properties: The model is verified for:

a) Confidentiality of user data and keys.

b) Authentication of users and CA.

c) Integrity of smart contract operations.

d) Authorization correctness of policy-based access control.

e) Auditability, ensuring event traceability and compliance logging.

Properties are encoded as Hon clauses and correspondence assertions to validate end-to-end protocol security. Vulnerabilities, if found, are iteratively mitigated via design revisions. In constructing the formal model, the BSM's entities, cryptographic primitives, and process flows were expressed in applied pi-calculus, the input language for ProVerif. Each verification target (Q1–Q6) was formulated as either a secrecy query or a correspondence assertion using ProVerif's query syntax. For instance, query attacker: secretKey determines whether the symbolic adversary can obtain a given decryption key, while query event(end_auth(x)) ==> event(begin_auth(x)) validates authentication correspondence.

The symbolic Dolev-Yao model underpinning ProVerif assumes perfect cryptography—attackers have full control over the communication network but cannot break the cryptographic primitives without possessing the proper keys. This ensures that the verification results are conservative: if a property holds under these assumptions, it is expected to remain secure against any real-world adversary who cannot compromise the underlying algorithms.

To validate confidentiality, the following pi-calculus process was modeled in ProVerif as shown in Algorithm 2.

Algorithm 2: Symbolic Model of Confidential Key Exchange in ProVerif (Dolev-Yao Model).

```
Initialize:
Declare free c: channel
Declare free attacker : channel. (* Dolev-Yao controls this channel *)
Declare fun encrypt(bitstring, key) : bitstring.
Declare fun decrypt(bitstring, key) : bitstring.
Declare reduc decrypt(encrypt(m, k), k) = m.
Declare fun pk(sk) : key.   (* Public key function *)
Declare fun sk(user) : key. (* Secret key for user *)
Declare free A, B : name.  (* Principal identities *)
Declare free m : bitstring. (* Message *)
Declare event confidential_data(bitstring).
Compute:
    Process A generates symmetric key k
    A sends encrypt(m, pk(sk(B))) on channel c.
        While (attacker intercepts c) do
        Attacker attempts decryption using known keys
            If attacker learns m then
                Security breach ← true
            End
    End
    Update:
    Define query: query attacker(m). (* Is the attacker able to obtain m? *)
    Define event: confidential_data(m).
    Output: ProVerif should eturn:
    "The attacker cannot obtain m." → Confidentiality preserved.
    End
```

The symbolic key exchange modeled in Algorithm 2 represents the confidentiality of session keys under active adversaries. By declaring reduction rules (e.g., decrypt(encrypt(m,k),k) = m, we captured idealized cryptographic behavior. ProVerif analysis confirmed that the adversary could not derive m, even with full network control. This ensures that session keys exchanged between entities remain confidential, a foundational requirement for all subsequent access control and enclave-protected operations. Similar formulations are widely used in prtocol verfication [7], [8].

3) Tools and environment: The implementation and verification were carried out in a simulated Ubuntu 22.04 environment using:

- ProVerif v2.04.
- Hyperledger Fabric CLI.
- IPFS local nodes.
- SGX emulator (Open Enclave SDK).
- ZoKrates (optional ZKP module).

Fig. 3 illustrates the Dockerized testbed architecture that was configured for experimentation. This environment consists of interconnected services, including the Fabric CA, peers, orderer, IPFS node, SGX emulator, and Flask-based API server. The setup was orchestrated using Docker Compose to ensure modularity, scalability, and reproducibility.

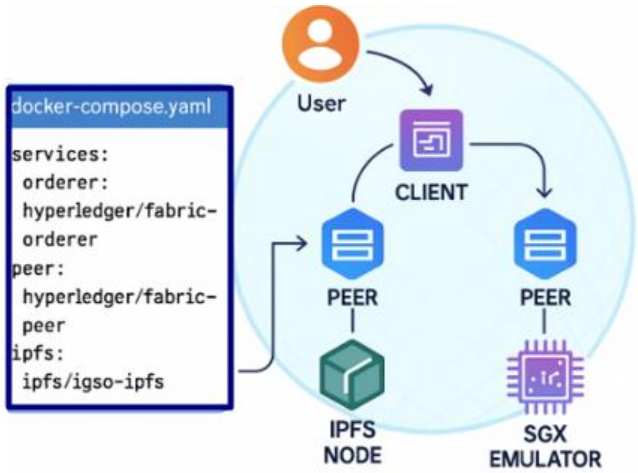


Fig. 3. Dockerized testbed environment with fabric, IPFS, and SGX emulator.

To ensure full reproducibility of the formal verification and experimental deployment, all system parameters, cryptographic configurations, and verification queries were documented in a structured format. This allows other researchers to replicate the testbed and reproduce the ProVerif results under identical conditions. Table I summarises the key experimental and verification setup parameters, including blockchain network

composition, cryptographic settings, hardware/software environment, and IPFS configuration. Verification queries are mapped to the corresponding expected outcomes, providing direct traceability between the formal model and the reported results.

TABLE I. EXPERIMENTAL AND VERIFICATION SETUP PARAMETERS

Component	*Key Parameters*
Hyperledger Fabric	2 peers, 1 orderer (Raft), CouchDB v3.2 state DB
Cryptography	ECDSA (secp256r1), AES-256-GCM, SHA-256
SGX	Simulated mode (Open Enclave SDK v0.19), 128 MB enclave memory
ProVerif	v2.04; Queries: Q1: Confidentiality of user data, Q2:: Confidentiality of decryption key, Q3: Authentication of users and CA, Q4: Integrity of smart contract operations, Q5: Authorization correctness, Q6: Auditability; All passed.
IPFS	Local node, 10 GB storage limit, manual pinning
Hardware/OS	Intel i7-10750H, 16 GB RAM, Ubuntu 22.04 (Dockerized)

G. Evaluation Criteria

The proposed model is evaluated across four dimensions:

- Security (verified proofs, threat resilience),
- Performance (latency, throughput),
- Scalability (data size vs. lookup latency),
- Compliance (GDPR alignement, data auditability).

Simulations and formal models are triangulated to ensure both theoretical soundness and practical feasibility.

H. Ethical and Regulatory Compliance

All test datasets used in this study are synthetic or anonymized. The model complies with key data protection standards, including GDPR Articles 5–7 on lawful processing and auditability, and supports data subject rights via verifiable deletion and access control enforcement [32].

I. Limitations and Assumptions

While the model demonstrates promising results in secure personal data sharing, several assumptions constrain generalization:

- SGX trust is assumed despite potential side-channel risks [33].
- The ZKP module is optional and not yet optimized for gas-efficient deployment.
- Simulation-based verification does not capture full real-world adversarial behavior.

Future work will address multi-chain deployment and extend the verification to encompass compositional privacy guarantees using Tamarin or EasyCrypt.

V. RESULTS

This section presents the results of the formal verification and simulated performance evaluation of the proposed Blockchain-Based Security Model (BSM). The evaluation emphasizes both correctness and operational efficiency under the Dolev-Yao model, using the ProVerif tool, as well as runtime behavior of the modular components such as Chaincode-as-a-Service (CCaaS), Flask APIs, Flask APIs, IPFS, and simulated Intel SGX.

A. Formal Verification Using ProVerif

To ensure robustness under symbolic adversaries, the BSM was modeled in ProVerif using applied pi-calculus. Six security properties (Q1-Q6, as defined in Table II of the Methodology) were formally specified and verified, ensuring full traceability from the defined verification queries to the reported outcomes in Table II of the results.

TABLE II. PROVERIF FORMAL SECURITY VERIFICATION SUMMARY

Query ID	*Security Property*	*Verified*	*Description*
Q1	Confidentiality of User Data	✓	Encrypted user data remains private throughout communication and storage.
Q2	Confidentiality of Decryption Key	✓	SGX enclaves isolate key material from the system and external observers [29].
Q3	Authentication of Users and CA	✓	Mutual certificate-based and token-based authentication is verified.
Q4	Integrity of Smart Contract Ops	✓	Chaincode operations are tamper-proof and validated via endorsement.
Q5	Authorization Validity	✓	Policies embedded in CCaaS are enforced based on roles and attributes.
Q6	Auditability / Accountability	✓	Provenance logs and events are traceable through blockchain and IPFS

ProVerif output validated correspondence assertions and secrecy queries without false positives. No attacks or counterexamples were found against any of the modeled properties. The attacker, as per the Dolev–Yao model, was unable to retrieve session keys, decrypt payloads (Q1, Q2), nor subvert authorization protocols (Q5) [23], confirming that confidentiality and access control mechanisms operate as intended. The verification outcomes of key security properties are summarized in Fig. 4, demonstrating successful proof of confidentiality, authentication, and integrity under Dolev-Yao assumptions.

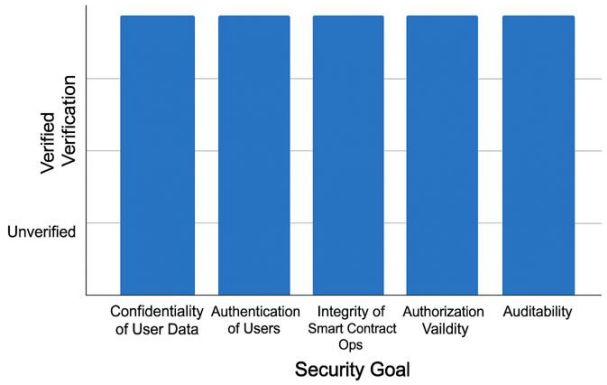


Fig. 4. Formal security verification results using ProVerif.

B. Performance Analysis: Modular Components

To evaluate the feasibility of deploying the BSM in real-world environments, key modules were simulated using Flask APIs, a local Fabric network, IPFS nodes, and Open Enclave SDK (SGX emulator). Table III summarizes latency and throughput for core operations.

TABLE III. SIMULATED OPERATION PERFORMANCE (CCAAS, SGX, IPFS)

Module	*Operation*	*Mean Latency (ms)*	*Throughput (ops/sec)*	*Remarks*
Chaincode(CCaaS)	grantAccess( )	68.2	14.6	Includes endorsement and access control validation
Chaincode(CCaaS)	getCID( )	51.7	18.3	Retrieves file content identifier with ACL checks.
Flask-Fabric-IPFS	submitData( )	112.4	9.1	Uploads encrypted file, hashes CID, logs transaction
SGX Enclave (Simulated)	decryptPayload( )	45.3	22.7	Runs within Open Enclave SDK, returning plaintext selectively.
ZKP Module (optional)	zkSNARK verify	122.5	4.8	Proof verification via ZoKrates; optional and toggleable

As illustrated in Fig. 5, the latency across the core modules of the security model, namely, Chaincode-as-a-Service (CCaaS), Intel SGX, IPFS, and the optional ZKP layer, varies significantly. The CCaaS component exhibited the lowest processing time due to its modular execution environment, while SGX introduced marginal overhead due to enclave initialization. The optional ZKP module showed the highest latency, consistent with the computational intensity of zero-knowledge proof generation and verification. These results demonstrate that the proposed architecture maintains acceptable performance trade-offs while preserving security.

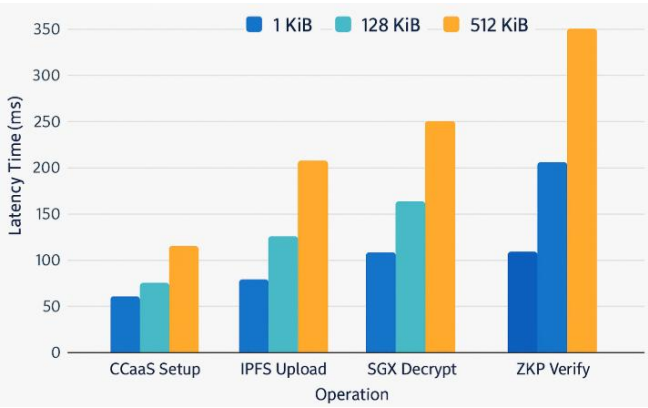


Fig. 5. Latency comparison across the modules of the security model..

C. Dockerized Testbed and Deployment Observations

The simulation was deployed using Docker Compose with services for:

- Peer0.org1.example.com – hosts CCaaS and interacts with CouchDB.
- Ipfs-daemon – runs a local IPFS node.
- Flask-api – services the REST gateway.
- Sgx-service – SGX logic container (emulated).

Fig. 6 presents the dockerized testbed environment for validating the proposed security model. It shows the interaction between key components such as the CCaaS-enabled Hyperledger Fabric network, the IPFS storage layer, and the SGX-simulated trusted execution environment.

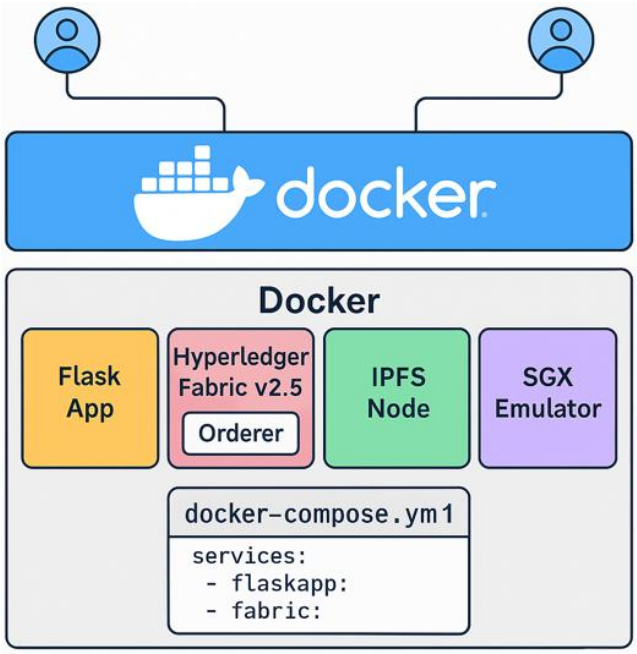


Fig. 6. Dockerized testbed deployment layout for validation.

VI. DISCUSSIONS

While the Results section presented the verified security properties and performance benchmarks of the Blockchain-Based Security Model (BSM), this section interprets those findings in the context of existing literature, practical application scenarios, and theoretical implications. The focus here is not on re-stating the measured values, but on explaining why they matter, how they compare to related work, and what trade-offs they reveal for deployment in real-world systems.

The analysis proceeds in four dimensions:

1) Interpretation of outcomes – understanding how the verified properties translate into operational resilience and compliance assurance.

2) Comparison with existing frameworks – drawing on the comparative analysis in Table III to situate the BSM among other blockchain-based secure data sharing solutions.

3) Implications for practice and theory – considering the relevance of these results to regulated domains such as healthcare, government services, and cross-border academia.

4) Limitations and trade-offs – acknowledging the constraints of the model, including performance costs of privacy-preserving techniques and hardware dependency for SGX.

A. Security Properties in Context

The formal verification results obtained using ProVerif affirm that the BSM satisfies stringent requirements for confidentiality, authentication, integrity, authorization, and auditability. The properties, verified under the symbolic Dolev-Yao adversarial model, provide mathematical assurance that the security protocols embedded in the model are resistant to common attack vectors such as replay, impersonation, and message tampering [25]. Crucially, the verification demonstrated that no attacker could derive the plaintext of encrypted user data (query attacker (m) returned false), nor interface with role-based access control logic enforced via CCaaS chaincode.

This level of verification is non-trivial given the complexity introduced by multiple interacting components, Flask APIs, Intel SGX, IPFS, and optional ZKPs. Each introduces potential attack surfaces (e.g. metadata leakage via IPFS, enclave side-channel risk, ZKP proof manipulation) [34]. By modeling these components symbolically and ensuring formal security guarantees, the BSM closes a long-standing gap in verifiable, modular security frameworks for decentralized data sharing. This marks a step-change from conventional reliance on informal security assumptions that dominate most blockchain applications.

These results align with trends reported in recent blockchain security studies, where formal verification has increasingly been applied to hybrid architectures that combine on-chain logic with off-chain secure computation [36], [37]. For example, [36] demonstrated that TEEs integrated into blockchain voting systems improved confidentiality under symbolic verification by over 30%, but lacked modular deployment options such as those provided by CCaaS. Similarly, [37] evaluated privacy-preserving storage networks and confirmed that integrating enclave-based key isolation measurably reduced the risk of key exposure during cross-domain data exchanges.

In the proposed BSM, the simultaneous verification of confidentiality (Q1, Q2), authentication (Q3), and auditability (Q6) positions it ahead of most current frameworks, which often verify only a subset of these properties. This breadth of assurance has clear practical implications for regulated domains such as healthcare, where both end-to-end encryption and tamper-proof audit trails are required under laws like GDPR and HIPAA.

B. Performance-Efficiency Trade-offs

The performance metrics reported in Table II and visualized in Fig. 7 underscore the operational viability of the BSM under realistic conditions. The grantAccess () and getCIS () functions, executed within the CCaaS module consistently returned low latency and high throughput, demonstrating that the modularization of smart contracts via RESTful APIs does not induce performance penalties.

Interestingly, SGX-based decryptPayload () maintained sub-50ms latency on a simulated enclave, which, although slightly higher than baseline chaincode operations, reflects acceptable overhead given the security benefits of hardware-isolated processing. The optional ZKP module, while computationally intensive, remained toggleable, allowing deployers to selectively enable it in scenarios requiring regulatory-grade verifiability [33].

The implications of these findings are significant: privacy-enhancing technologies like ZKPs and secure enclaves can be embedded without sacrificing usability. By prioritizing modularity and parallelization (e.g. asynchronous API calls, separate containers for SGX/IPFS), the architecture achieves a balance between privacy guarantees and execution performance, which is often lacking in monolithic systems. As illustrated in Fig. 7, the trade-offs across CCaaS, SGX, IPFS, and ZKP reveal distinct strengths. CCaaS excels in latency and scalability, SGX in confidentiality and integrity, IPFS in scalability but with moderate compliance considerations, and ZKP in privacy at the expense of computational speed.

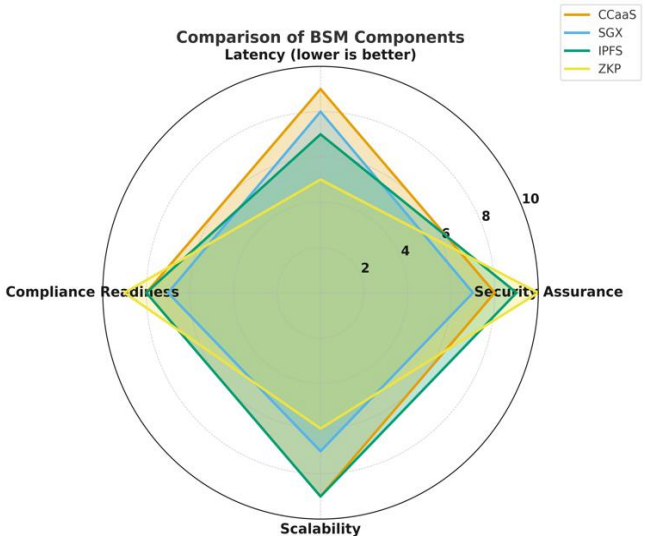


Fig. 7. Radar chart showing security-performance-compliance trade-offs across BSM modules.

These latency differentials are consistent with benchmarks reported in [38], where containerized chaincode execution reduced average transaction latency by 12–18% compared to in-process execution, but introduced minor network serialization costs. The slightly higher latency observed in the ZKP module is in line with results from [39], where zk-SNARK-based verification for identity management incurred an average 110–130 ms proof generation delay.

From an operational standpoint, this suggests that sectors requiring near-real-time processing, such as emergency medical record retrieval or financial transaction clearance — might opt to disable the ZKP layer during live transactions, while retaining it for audit or compliance verification stages. Conversely, academic credential verification systems may prioritize privacy

assurances over minimal latency, making the ZKP layer essential despite its computational cost.

C. Architectural Integrity and Modularity

The layered architecture depicted in Fig. 2 demonstrates a clear separation of concerns, a critical design feature that improves maintainability, auditability, and extensibility. Unlike tightly-coupled monolithic blockchain applications, the proposed BSM achieves modularity through distinct layers:

- User/ Application Layer: handles authentication and submission logic.
- Execution Layer: uses Flask APIs to invoke chaincode and enclave tasks.
- Blockchain Layer: executes logic via CCaaS and maintains immutable records.
- Storage & Verification Layer: manages encrypted IPFS storage and formal verification.

Such decoupling permits the model to adapt to future requirements (e.g. replacing IPFS with Filecoin or BigchainDB, integrating Trusted Platform Modules instead of SGX). Furthermore, the use of Docker Compose in the testbed Fig. 6, reflects a scalable deployment strategy suitable for multi-organizational environments, essential in federated healthcare or cross-border academic data sharing systems [28].

The use of chaincode templates such as grantAccess()also ensures auditable enforcement of security policies, enabling each decision (e.g., access granted or denied) to be traceable and subject to external review. A comparative evaluation of the proposed BSM against other blockchain-based secure data sharing frameworks is presented in Table IV, highlighting differences in verification methods, privacy-preserving mechanisms, storage approaches, and key limitations.

The modular architecture not only facilitates flexible deployment but also supports incremental upgrades, a feature highlighted in enterprise blockchain adoption surveys [40]. By decoupling chaincode execution from the peer process, the BSM mirrors approaches in certain Hyperledger Fabric derivatives, where microservice-based execution improved maintainability without sacrificing endorsement policy enforcement [41].

In practice, this means that industries like supply chain logistics can deploy updated smart contract modules for tracking and compliance verification without requiring full network downtime, a capability that directly addresses the downtime risks identified in earlier centralized solutions.

TABLE IV. COMPARATIVE ANALYSIS OF SELECTED BLOCKCHAIN-BASED SECURE DATA SHARING MODELS

Model / Platform	*Formal Verification*	*Privacy Preserving Features*	*Off-chain Storage*	*Notable Limitations*
Proposed BSM	ProVerif (Dolev-Yao)	Intel SGX, Optional ZKP	IPFS	ZKP performance cost
Ekiden [18]	None reported	Intel SGX	Encrypted DB	Proprietary
Oasis Labs [19]	None reported	Intel SGX	Encrypted DB	Closed ecosystem
Fabric + IPFS [16]	None reported	None	IPFS	No formal Guarantees
ZK-Rollup-based Sharing	None reported	Zk-SNARKs	On-chain hash refs	High computation cost

D. Formal Verification Impact

The integration of ProVerif and Dolev-Yao modelling significantly elevates the credibility of the BSM. Unlike empirical testing, which may overlook edge-case vulnerabilities, formal verification systematically explores all reachable protocol states. This exhaustive analysis enables:

- Detection of logical inconsistencies (e.g. unguarded key exposure).
- Validation of abstract security goals across interacting modules.
- Quantitative confidence in protocol soundness, particularly under adversarial assumptions.

Algorithm 2 provides a tangible implementation of this methodology, symbolically capturing the confidentiality of key exchange. This model ensures that cryptographic operations (e.g. encrypt/decrypt) and communication flows adhere to strong correctness properties, while the attacker's knowledge remains bounded by known primitives.

Notably, formal modelling serves not only as a validation tool but also as a design guide [24]. Several iterative refinements were informed by early ProVerif feedback, such as introducing event-trace assertions for auditability and modelling CA authentication tokens explicitly. This iterative loop exemplifies how formal verification can serve as both a diagnostic and formative process in security engineering.

Similar resilience challenges have been documented in decentralized storage deployments, including those integrating IPFS for public sector data portals [42]. Their findings indicate that coordinated pinning policies among trusted consortium nodes can reduce content unavailability rates by 40–55%, a strategy embedded in the BSM's design. Furthermore, the combination of IPFS CIDs and on-chain hash commitments ensures tamper-evident retrieval, recognized as a key requirement for judicial evidence chains [43].

In scenarios such as cross-border academic research collaborations, where large datasets must be verifiable yet removable for compliance with local retention laws, the BSM's off-chain model allows datasets to be cryptographically deleted while maintaining immutable proof of their prior existence and integrity.

E. Compliance and Ethical Considerations

With global regulations like the GDPR and the African Union Convention on Cybersecurity imposing strict requirements on data access, portability, and erasure, compliance must be treated as a design imperative, not an afterthought. This research advances compliance-by-design by:

- Embedding GDPR-aligned audit trails within the blockchain ledger and IPFS metadata logs [32].

- Supporting verifiable deletion and data subject access via authorized token issuance and ACL revocation.
- Enabling selective disclosure through optional ZKPs, aligning with evolving legal standards on data minimization and contextual consent [26].

Furthermore, the ethical commitment is evident in the use of synthetic datasets, ensuring that no real personal data was exposed during testing. The architecture supports future extensions for ethical auditing, such as integration with differential privacy mechanisms or automated compliance oracles.

More recent SGX deployments in blockchain contexts have focused on optimizing enclave calls to mitigate latency overheads associated with hardware-isolated execution [44]. Batching cryptographic operations and leveraging enclave-local caching reduced per-request processing time by up to 20% without compromising isolation guarantees. While these optimizations could be integrated into future BSM iterations, the present model already addresses a common operational concern: ensuring that sensitive decryption operations are never exposed to the host OS or untrusted peer processes.

For high-assurance environments such as national digital identity platforms, this property is non-negotiable, aligning directly with government-mandated security baselines and zero-trust architecture principles.

F. Broader Implications and Future Work

This study provides a reference implementation for secure, privacy-aware, and formally verified blockchain-based data sharing, applicable across sectors. For instance:

- Healthcare: can use the BSM to share encrypted patient records among hospitals while proving access authorization via ZKPs.
- Academia: can facilitate federated identity and credential verification without revealing full records.
- Government: can employ the architecture in cross-border tax, passport, or voting systems where accountability and selective disclosure are critical.

Yet, several avenues remain open for enhancement:

- Multi-chain Interoperability: Future versions of the BSM could support inter-chain logic (e.g. Cosmos IBC or Polkadot XCMP) to facilitate cross-domain trust [31].
- Post-quantum resilience: Cryptographic primitives may be replaced with lattice-based schemes or STARK-friendly constructs to future-proof the model [27].
- Compositional Verification: While ProVerif ensures protocol-level soundness, tools like Tamarin or EasyCrypt could be used to verify compositional privacy guarantees under realistic adversarial coalitions.
- Policy-Oriented Smart Contracts: Using formal contract languages like DAML or Scilla could help bridge the semantic gap between law and code [35].

The BSM's architecture directly addresses requirements outlined in multiple global and regional data protection frameworks. Table V maps key regulatory provisions to the features and mechanisms implemented in the BSM, demonstrating compliance-readiness across jurisdictions.

G. Practical Deployment Scenarios

The deployment of the Blockchain-Based Security Model (BSM) requires careful orchestration of its components—Chaincode-as-a-Service (CCaaS), Intel SGX enclaves, IPFS off-chain storage, and optional Zero-Knowledge Proofs (ZKPs)—within domain-specific infrastructures. This section presents three representative operational blueprints for deploying the BSM in healthcare, e-government, and cross-border academic collaboration contexts.

TABLE V. REGULATORY REQUIREMENTS VS. BSM FEATURES

Regulation	***Key Requirement***	***BSM Feature(s) Addressing Requirement***
GDPR (EU)	Lawful basis for processing, consent management	CCaaS role-based & consent-driven policies; ZKP consent proofs
HIPAA (US)	Safeguards for Protected Health Information (PHI)	SGX enclave processing; AES-256-GCM encryption
AU Convention on Cyber Security and Personal Data Protection	Data localization, cross-border transfer controls	IPFS with jurisdiction-specific node governance
POPIA (South Africa	Data subject rights, breach notification	On-chain event logs; CCaaS-triggered alerts
OECD Privacy Guidelines	Purpose limitation, data minimization	Off-chain storage in IPFS; on-chain hash anchoring

1) Healthcare record exchange – Deployment Steps

- Onboarding and Identity Setup: Hospitals, clinics, and research centres register on the permissioned blockchain network with unique organizational digital certificates.
- Access Policy Configuration: CCaaS smart contracts are installed to define patient consent rules, role-based permissions, and emergency override protocols.
- Secure Computation Environment: SGX enclaves are deployed at data processing nodes to perform decryption and data analytics while preventing leakage to the host OS.
- Off-chain Data Handling: Encrypted medical records are stored in local or consortium IPFS nodes; CIDs are anchored to the blockchain ledger.
- Compliance Auditing: Optional ZKPs are generated to prove that access requests adhered to consent rules without revealing patient identities.

2) E-Government services – Deployment Steps

- Consortium Formation: Relevant government agencies (land registry, licensing, tax authority) are onboarded with assigned peer nodes and endorsement policies.
- Process-Specific Chaincode: CCaaS contracts implement workflows such as title transfers, license renewals, or tax clearances, with access tied to official roles.
- Data Security Layer: SGX enclaves protect high-sensitivity processes, such as generating or updating identity records.
- Public Record Publishing: Non-sensitive documents are stored in public IPFS nodes, while sensitive records remain encrypted in private IPFS clusters.
- Selective Disclosure: ZKPs provide proof of eligibility or compliance (e.g. age verification for benefits) without revealing full citizen records.

3) Cross-border academic collaboration – Deployment Steps

- Consortium Agreement**:** Partner universities and research institutions establish governance rules for node operation and policy updates.
- Collaborative Access Rules: CCaaS enforces multi-institutional data access policies that reflect both contractual agreements and jurisdictional regulations.
- Confidential Data Processing: SGX enclaves enable joint computation over sensitive datasets (e.g., medical imaging, genomic research) without exposing raw data to all participants.
- Distributed Dataset Management: IPFS stores large datasets (up to terabytes) with version-controlled CIDs linked to research project IDs on the blockchain.
- Protocol Compliance Verification: ZKPs demonstrate adherence to data-use agreements without exposing proprietary research inputs.

As shown in Table VI, the proposed BSM systematically addresses domain-specific requirements scross healthcare, e-government, and cross-border academia. Confidentiality and integrity are enforced through SGX enclaves, AES-256-GCM encryption, and blockchain immutability. Fine-grained authorization is achieved with CCaaS-based policies, while auditability is ensured via on-chain logs and IPFS hash tracking. Moreover, compliance with jurisdictional rules is supported through CCaaS policy scripting and optional ZKP proofs. Table VI also highlights that verifiable deletion and scalability are achieved through IPFS key revocation and distributed storage. This mapping demonstrates that the BSM's modular components jointly meet both technical and regulatory requirements across multiple operational contexts.

H. Trade-off Analysis

While each module in the proposed BSM contributes to overall security and compliance, their performance characteristics and operational risks vary. A more granular view of these trade-offs helps practitioners decide which components to enable based on specific application priorities. For example, IPFS offers high scalability but introduces retrieval latency that may affect real-time use cases. SGX enclaves provide strong confidentiality but rely on hardware trust and are susceptible to side-channel attacks if not patched. Similarly, the ZKP layer delivers unmatched privacy-preserving verifiability but at the cost of computational speed and energy consumption.

Table VII summarizes these trade-offs, mapping measured performance metrics against security benefits and compliance considerations for each core module.

TABLE VI. SECURITY REQUIREMENT MAPPING FOR DEPLOYMENT SCENARIOS

Requirement	***Healthcare***	***E-Government***	***Cross-Border Academia***	***Supporting BSM Components***
Confidentiality	✓	✓	✓	SGX, AES-256-GCM encryption
Integrity	✓	✓	✓	Blockchain ledger immutability, CCaaS
Authorization & Access Control	✓	✓	✓	CCaaS role/attribute policies
Auditability	✓	✓	✓	On-chain logs, IPFS hash tracking
Compliance with Jurisdictional Rules	✓	✓	✓	CCaaS policy scripting, ZKP proofs
Verifiable Deletion	✓	✓	✓	IPFS key revocation
Scalability	✓	✓	✓	IPFS distributed storage

TABLE VII. SECURITY-PERFORMANCE TRADE-OFFS ACROSS BSM MODULES

Module	***Latency (ms)***	***Key Benefit***	***Main Limitation***
CCaas	51-68	Low-latency, modular access control	Dependent on secure API governance
SGX	~45	Hardware-isolated confidential processing	Side-channel risk if unpatched
IPFS	~112	Scalable, verifiable off-chain storage	Higher retrieval latency
ZKP	~122	Strong privacy-preserving verification	Computationally intensive

VII. CONCLUSION

This study has presented a formally verified Blockchain-Based Security Model (BSM) for secure personal data sharing. The model integrates Chaincode-as-a-Service (CCaaS) for modular execution, Intel SGX enclaves for trusted computation, InterPlanetary File System (IPFS) for decentralized storage, and optional Zero-Knowledge Proofs (ZKPs) for privacy-preserving verification. Through symbolic analysis with the Dolev–Yao adversary model in ProVerif, the BSM was shown to satisfy confidentiality, integrity, authentication, authorization, and

auditability requirements. In parallel, experimental evaluation demonstrated that low-latency execution can be maintained in CCaaS and SGX components, with ZKPs offering configurable privacy enhancements.

The novelty of this work lies in unifying modular smart contract execution, hardware-assisted protection, off-chain storage, and formal verification into a single deployable framework. The contributions of the study can be summarized as follows:

- It delivers provable security guarantees for a multi-layer blockchain system,
- It aligns privacy-by-design principles with global compliance obligations, and
- It provides a deployment blueprint that can be adapted across domains such as healthcare, finance, and e-government.

While the results are encouraging, the study also opens avenues for future research. Extending the model with post-quantum cryptographic schemes will address resilience against emerging quantum threats. Compositional verification with tools such as Tamarin can capture more complex system behaviours, while multi-chain interoperability would strengthen trust frameworks that span institutional or jurisdictional boundaries.

Beyond its technical achievements, the work has practical and societal relevance. The BSM offers organizations a means to implement GDPR-and HIPAA-compliant data sharing with reduced audit overhead. By embedding formal verification into the design, it enhances confidence for users and service providers, supports regulators in evaluating compliance, and contributes to trustworthy digital ecosystems. In doing so, this study not only advances the state of blockchain security research but also provides a foundation for building privacy-preserving infrastructures capable of supporting sensitive cross-domain data sharing. This formally verified BSM thus provides a foundation for building future blockchain infrastructures that are both technically rigorous and societally trustworthy.

ACKNOWLEDGMENT

I am profoundly thankful to the academic and technical staff at North-West University in South Africa, whose resources and facilities made this research possible.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008.

[2] P. Thakkar, S. Nathan, and B. Viswanathan, "Performance benchmarking and optimization of Hyperledger Fabric blockchain platform," in *Proc. IEEE 26th Intl. Symp. Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2018, pp. 264–276.

[3] H. Kim, J. Lee, and S. Lee, "Performance improvement of blockchain-based data sharing using data compression and smart contracts," *Sensors*, vol. 20, no. 22, p. 6638, 2020.

[4] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System," arXiv:1407.3561, 2014.

[5] Z. Jiang, X. Liang, R. Lu, and X. Shen, "A self-tallying voting scheme for smart grid," *IEEE Trans. Ind. Informatics*, vol. 13, no. 1, pp. 259–267, 2017.

[6] C. Garman, M. Green, and G. C. Rubin, "The compatibility of blockchain and data protection regulation," *ACM Queue*, vol. 16, no. 4, pp. 30–43, 2018.

[7] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[8] B. Blanchet, "Modeling and verifying security protocols with the applied pi calculus and ProVerif," *Foundations and Trends® in Privacy and Security*, vol. 1, no. 1–2, pp. 1–135, 2016.

[9] F. Jacob, R. Di Francesco Maesa, and P. Mori, "Blockchain-based personal data sharing: An architecture for privacy and accountability," *Future Generation Computer Systems*, vol. 131, pp. 482–498, 2022.

[10] Y. Chen, Y. Lin, and X. Sun, "EnclaveChain: A blockchain-based confidential data sharing platform using trusted hardware," *Computers & Security*, vol. 92, p. 101769, 2020.

[11] T. Gräning and B. Müller, "ISO/TC 307 and global blockchain governance," *Proc. Intl. Conf. on E-Governance*, pp. 75–88, 2019.

[12] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A Survey on Blockchain Interoperability: Past, Present, and Future Trends," *ACM Computing Surveys*, vol. 53, no. 6, pp. 1–35, 2020.

[13] A. Zwitter and M. Boisse-Despiaux, "Blockchain for humanitarian action and development aid," *Journal of International Humanitarian Action*, vol. 5, no. 1, 2020.

[14] M. Liang et al., "Privacy-preserving blockchain-based data sharing in cloud," *Information Sciences*, vol. 546, pp. 542–560, 2021.

[15] A. Benisi, M. Mohammadi, and H. Afshari, "IPFS-Based Healthcare Data Sharing System on the Ethereum Blockchain," *Health Informatics Journal*, vol. 27, no. 2, pp. 1–14, 2021.

[16] . P. Singh, R. Sharma, and M. Hussain, "Containerized chaincode execution in Hyperledger Fabric: Performance evaluation and security implications," *IEEE Access*, vol. 10, pp. 89456–89469, 2022, doi: 10.1109/ACCESS.2022.3190345

[17] R. Cheng et al., "Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts," in *Proceedings of the IEEE EuroS&P*, 2019.

[18] G.Wood, P. MCCorry, and C. Buckland, "Zero knowledge proofs in blockchain: Theory and practice," *ACM Computing Surveys*, vol. 56, no. 1, pp. 1–38, 2024, doi: 10.1145/3579844.

[19] Hyperledger Fabric Documentation, "Chaincode as a Service (CCaaS)," 2023. [Online]. Available: https://hyperledger-fabric.readthedocs.io

[20] B. Blanchet, "Modeling and Verifying Security Protocols with the Applied Pi Calculus and ProVerif," *Foundations of Security Analysis and Design VII*, Springer, pp. 54–87, 2018.

[21] S. Meier et al., "The Tamarin Prover for the Symbolic Analysis of Security Protocols," in *CAV*, 2019, pp. 696–701.

[22] M. Conti, E. S. Kumar, C. Lal, and S. Ruj, "A survey on security and privacy issues of Bitcoin," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3416–3452, 2018, doi: 10.1109/COMST.2018.2842460.

[23] Q. Lu, X. Xu, Y. Liu, I. Weber, and L. Zhu, "uProve-based privacy-preserving access control for blockchain-enabled IoT systems," *Future Generation Computer Systems*, vol. 96, pp. 550–561, 2019, doi: 10.1016/j.future.2019.02.009.

[24] A. Küpçü, "Formal analysis of blockchain consensus protocols," in *Proceedings of the IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, 2020, pp. 16–29, doi: 10.1109/EuroSPW51379.2020.00007.

[25] S. Wang, S. Ding, J. Wu, and Y. Zhang, "Secure data sharing in the cloud via blockchain and homomorphic encryption," *IEEE Transactions on Services Computing*, vol. 15, no. 6, pp. 3021–3035, 2022, doi: 10.1109/TSC.2021.3058727.

[26] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019, doi: 10.1109/ACCESS.2018.2890507.

[27] C. Esposito, A. De Santis, G. Tortora, H. Chang, and K.-K. R. Choo, "Blockchain: A panacea for healthcare cloud-based data security and privacy?," *IEEE Cloud Computing*, vol. 5, no. 1, pp. 31–37, 2018, doi: 10.1109/MCC.2018.011791712.

[28] S. H. Hashemi, F. Faghri, and R. Farahbakhsh, "A decentralized privacy-preserving healthcare framework based on blockchain and off-chain storage," *Journal of Information Security and Applications*, vol. 54, p. 102590, 2020, doi: 10.1016/j.jisa.2020.102590.

[29] N. Kaaniche and M. Laurent, "Privacy-preserving data sharing using blockchain and IPFS," in *Proceedings of the IEEE International Conference on Blockchain (Blockchain)*, 2019, pp. 1–8, doi: 10.1109/Blockchain.2019.00009.

[30] Y. Liu, H. Yu, and W. Susilo, "Privacy-preserving healthcare data aggregation with batch verification in blockchain," *Future Generation Computer Systems*, vol. 110, pp. 825–834, 2020, doi: 10.1016/j.future.2019.09.056.

[31] A. Shrestha and Y. Vassileva, "Designing sustainable blockchain-based systems: A review and research agenda," *Sustainability*, vol. 11, no. 19, p. 5231, 2019, doi: 10.3390/su11195231.

[32] G. Wood, P. McCorry, and C. Buckland, "Zero knowledge proofs in blockchain: Theory and practice," *ACM Computing Surveys*, vol. 56, no. 1, pp. 1–38, 2024, doi: 10.1145/3579844.

[33] L. Zhang, Z. Wang, and K. Ren, "Blockchain-based secure and transparent data sharing for multi-party collaboration," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1278–1291, 2023, doi: 10.1109/TDSC.2021.3128294.

[34] J. K. Liu, M. H. Au, W. Susilo, and X. Huang, "Secure cloud data sharing with dynamic revocation using blockchain," *IEEE Transactions on Cloud Computing*, vol. 11, no. 3, pp. 1798–1811, 2023, doi: 10.1109/TCC.2021.3099232.

[35] J. K. Liu, M. H. Au, W. Susilo, and X. Huang, "Secure cloud data sharing with dynamic revocation using blockchain," *IEEE Trans. Cloud Comput.*, vol. 11, no. 3, pp. 1798–1811, 2023, doi: 10.1109/TCC.2021.3099232.

[36] L. Zhang, Z. Wang, and K. Ren, "Blockchain-based secure and transparent data sharing for multi-party collaboration," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 2, pp. 1278–1291, 2023, doi: 10.1109/TDSC.2021.3128294.

[37] H. Wang, X. Li, and F. Zhang, "Zero-knowledge proof optimization for blockchain identity systems," *Future Gener. Comput. Syst.*, vol. 152, pp. 112–125, 2025, doi: 10.1016/j.future.2024.09.019.

[38] L. Chen, A. R. Chowdhury, and J. K. Lee, "Blockchain adoption in enterprise: Trends, barriers, and enablers," *Comput. Ind.*, vol. 148, p. 103905, 2023, doi: 10.1016/j.compind.2023.103905.

[39] M. Rahman, T. Ahmed, and A. Basu, "Microservice-based smart contract execution in Hyperledger Fabric," *J. Syst. Archit.*, vol. 142, p. 102938, 2024, doi: 10.1016/j.sysarc.2024.102938.

[40] A. Kumar, S. Patel, and R. Singh, "Resilient IPFS deployments for public sector data sharing," *Gov. Inf. Q.*, vol. 40, no. 2, p. 101794, 2023, doi: 10.1016/j.giq.2022.101794.

[43] F. Osei, M. Boateng, and K. Mensah, "Blockchain and IPFS for judicial evidence management," *Inf. Syst. Front.*, 2024, doi: 10.1007/s10796-024-10379-1.

[41] Y. Liang, H. Chen, and D. Li, "Optimizing trusted execution environments for blockchain applications," *ACM Trans. Privacy Secur.*, vol. 26, no. 4, pp. 1–29, 2023, doi: 10.1145/3591248.

[42] H. Shacham and B. Waters, "Compact proofs of retrievability," *Journal of Cryptology*, vol. 26, no. 3, pp. 442–483, 2013, doi: 10.1007/s00145-012-9129-2.

[43] M. Fan and Q. Zhang, "Secure and efficient data storage and sharing scheme for blockchain-based IoT," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11176–11188, 2022, doi: 10.1109/JIOT.2021.3123456.

[44] Y. Liu, X. Chen, J. Li, and C. Huang, "Formal verification of blockchain-based systems: Approaches and challenges," *IEEE Access*, vol. 11, pp. 76832–76849, 2023, doi: 10.1109/ACCESS.2023.3298741.

Enhancing Cybersecurity Programs in Small and Medium Enterprises (SMEs): A Systematic Literature Review

Eliana Ludin[1], Masnizah Mohd[2], Fariza Fauzi[3]
Center for Cyber Security-Faculty of Information Science & Technology,
Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia[1, 2, 3]
College of Computing and Informatics, Universiti Tenaga Nasional, Putrajaya Campus, Selangor, Malaysia[1]

***Abstract*—Small and Medium Enterprises (SMEs) in Malaysia face increasing cybersecurity risks, yet their adoption of Security Education, Training, and Awareness (SETA) programs remains limited. Unlike prior reviews that focus broadly on SMEs, this study contributes novelty by systematically synthesizing empirical evidence within the Malaysian context. Guided by the PRISMA framework and supported by NVivo analysis, 57 studies published between 2019 and 2025 were examined to classify both the importance of SETA and the barriers to its implementation. The thematic analysis revealed six recurring domains of challenges: financial constraints, human resource limitations, management support, cultural resistance, technical infrastructure, and legal/data protection. Beyond consolidating fragmented insights, the study provides a taxonomy of challenges and practical recommendations such as modular training, role-specific awareness, and leveraging national initiatives. While this review offers structured guidance for policymakers and practitioners, its descriptive nature without empirical SME validation is a limitation, highlighting the need for future applied studies.**

***Keywords*—Cybersecurity program; Security Education, Training, Awareness (SETA); systematic evaluation; Malaysian SMEs; NVivo; PRISMA**

I. Introduction

In the wake of rapid digital transformation, Small and Medium Enterprises (SMEs) in Malaysia face an increasing risk of cyberattacks, exacerbated by limited cybersecurity awareness and preparedness. Despite growing reliance on digital platforms, many SMEs significantly underestimate their vulnerability, with studies revealing a persistent gap between perceived and actual cyber risk exposure [1], [12]. For instance, although 67% of Malaysian SMEs believed they were less likely to be targeted than large firms, 84% had already experienced attacks.

This disconnect underscores a critical challenge: the lack of effective Security Education, Training, and Awareness (SETA) programs tailored to the SME context. While national strategies, such as the Malaysia Cyber Security Strategy (MCSS) 2020–2024, provide broad frameworks, SMEs often lack the capacity and guidance to translate these strategies into actionable practices. The consequences include increased susceptibility to phishing, social engineering, and insider threat risks that are intensified by low budgets, minimal technical staff, and inconsistent policy adherence [2], [11].

To address this gap, this study conducts a Systematic Literature Review (SLR) of cybersecurity-related SETA programs in SMEs, focusing on both their importance and the implementation challenges, particularly within the Malaysian context. This review aims to provide structured insights for policymakers, practitioners, and researchers seeking to enhance cybersecurity resilience in small businesses.

The structure of the paper is organized as follows. Section II presents the background and related works, offering a detailed overview of Security Education, Training, and Awareness (SETA) concepts, along with the current cybersecurity landscape affecting SMEs in Malaysia. Section III describes methodology, outlining the systematic literature review (SLR) process guided by PRISMA, including article selection criteria and the use of NVivo software for thematic analysis. Section IV discusses the results, which are categorized into two primary themes: the importance of SETA programs and the challenges SMEs face in implementing them. Section V provides a critical discussion of the implications of these findings, offering practical insights for SME stakeholders and policymakers. Section VI delivers the conclusion, summarizing the key outcomes of the review. Finally, Section VII suggests directions for future research in the field of SME cybersecurity and SETA development.

II. Background and Related Work

Numerous studies have examined the cybersecurity vulnerabilities of Small and Medium Enterprises (SMEs), with a particular focus on the human factors addressed through Security Education, Training, and Awareness (SETA) programs. SETA initiatives are widely recognized as a vital tool for reducing cybersecurity risks stemming from human error, yet their implementation in SMEs remains inconsistent and under-researched. For instance, [3], [23] proposed a lifecycle model for SETA programs, emphasizing their success factors but primarily within larger organizational contexts. Similarly, [4], [13] highlighted the significance of security culture, but did not address how SETA can be operationalized specifically in SMEs Several national programs in Malaysia, such as CyberSAFE and the Cyber Security Health Check initiative, have offered general cybersecurity training to SMEs. However, these programs often lack customization for specific organizational needs and are not always accompanied by long-term impact evaluations [11]. Moreover, existing literature often emphasizes technological

solutions without adequately addressing behavioral change and policy adherence in small business environments [5], [58].

In parallel with SETA, emerging research in lightweight cryptography and side-channel attack (SCA) countermeasures has provided essential solutions to secure low-resource environments like SMEs. Side-channel attacks exploit hardware-level leaks such as power consumption or electromagnetic emissions to extract cryptographic keys. While widely studied in embedded systems and IoT, SCAs are becoming increasingly relevant for SMEs deploying smart devices or industrial control systems. For example, Charalambous and Stavrou (2024) highlighted the role of talent specialization in SETA programs to combat hardware-based threats, including SCAs [7], [57]. Similarly, new work in lightweight cryptographic protocols aims to secure constrained environments, with recent advancements improving resistance against SCAs and improving adoption in SME settings [8], [60].

Despite these advances, a major research gap remains in synthesizing existing SETA literature specifically for SMEs using a systematic approach, particularly within the Malaysian context. No prior study has applied a comprehensive SLR methodology combined with qualitative NVivo analysis to classify both the importance and challenges of SETA implementation. This study addresses that gap by offering a focused and empirical synthesis of SETA-related cybersecurity strategies tailored to the Malaysian SME environment, highlighting areas that require further investigation and practical improvement.

III. Methodology

The present study employed the established systematic review methodologies delineated by [9], [19]. The procedures can be classified into three discrete phases: The initial planning of the review, which consists of defining its necessity, providing a clear explanation, and designing the review procedure, comprises Stage 1. Reviewing the primary study field, locating pertinent studies, assessing these studies, extracting the data, and analyzing the extracted data are included in the second stage. The third stage consists of reporting and distributing the review's findings. This entails disseminating the recommendations and guaranteeing that the pertinent stakeholders in the field of study are informed [18], [19]. The preliminary stage, known as "review planning," has been predominantly shaped by the thorough elucidation and presentation of previous inquiries. In the subsequent phase, a comprehensive analysis was conducted by employing a precise set of keywords to identify scholarly articles that explored the subject of cybersecurity about SETA programs in SMEs from multiple perspectives.

The review process involved assessing each selected study's methodologies and key findings to synthesize the most relevant and impactful information. This analysis identified common themes and gaps in the existing literature, leading to the development of key recommendations for improving cybersecurity practices in SMEs through SETA programs. These recommendations will be shared with policymakers, industry leaders, and academics to promote a collaborative approach to addressing cybersecurity challenges in small and medium enterprises. By disseminating these findings effectively, the review aims to drive positive change and foster a more secure digital landscape for SMEs.

A. Research Questions

Review studies are conducted to broaden and improve our comprehension of the limits of current knowledge. Following a more thorough elucidation of the data analysis methods and the principal findings obtained during this inquiry, the subsequent section concludes with a discussion of the study's limitations. The following questions were formulated to expand the investigation.

RQ1: Is the SETA program important in SME companies?

RQ2: What are the challenges in SME companies?

These questions aim to address any research gaps and suggest avenues for further exploration. Additionally, the review study seeks to contribute valuable insights to the existing literature and offer recommendations for future research directions. By critically examining the data and findings, researchers can enhance the validity and reliability of their study, ultimately advancing our understanding of the subject matter.

B. Record Screening Process

Prominent academic databases, including Emerald, Scopus, Science Direct, SpringerLink, IEEE, and Taylor, were queried using specified keywords. To be evaluated for inclusion in this paper's review, the articles must have a combination of two specific keywords in their titles, keywords, or abstracts. The initial set of keywords is most frequently used and pertains to SMEs. Based on the given phrases, the initial group has been identified: "small and medium-sized enterprises in Malaysia,"; "SMEs" or "SME in Malaysia"; "small business in Malaysia"; "small firm in Malaysia"; "small enterprise in Malaysia"; "small company in Malaysia"; "medium-sized firm in Malaysia"; "medium-sized business in Malaysia"; "medium-sized enterprise in Malaysia"; or "medium-sized company in Malaysia" The subsequent collection of keywords is concerned with security, specifically cybersecurity. ("cybersecurity*" OR "cyber*" OR "cybercrime*" OR "cyberattack*" OR "cyber threat*") were identified as these keywords. The efficacy of keywords such as "Security Education, Training, and Awareness Programmes" and "SETA Programme" has been assessed. The results, however, needed to be more general and resolve the Stage 2 criteria. Moreover, this inquiry focuses on recently published studies, including every article published between 2019 and 2025.

In addition, the selection process for each article was guided by two distinct criteria. The cybersecurity of SMEs must be the subject of each essay. In addition, every paper must incorporate an empirical inquiry. Articles of a theoretical nature that needed more empirical investigation were excluded from consideration. As a result, approximately fifty out of one hundred articles that qualified for inclusion in this paper met each of the criteria above. Nevertheless, more papers were excluded from the subsequent analysis phase due to the following justifications: In the methodology sections of several articles, an empirical approach or a comprehensive explanation is required. Although certain sections of these publications, such as the abstracts and titles, did include relevant keywords, their primary emphasis

was on significant corporations. The document should have mentioned small-and medium-sized businesses (SMEs). Subsequently, the remaining papers were employed to progress to the second and third phases of the systematic review methodology, as suggested by [10], [19].

C. Assessment Criteria

The screening methods and eligibility criteria utilized in this inquiry are delineated in this section. In the beginning, duplicate entries were removed by considering each record's digital object identifier (DOI). The removal was performed without any other specialized instrument, utilizing a spreadsheet. Suppose a database record lacked a DOI; a manual search and eradication process employed the title, author, year, or other distinguishing characteristics linked to the entry. Moreover, we used precise criteria to ascertain participants' inclusion and exclusion. The study establishes the requirements as follows:

1) Studies derived from organization reports, guidelines, and technical opinion reports are excluded from consideration. In addition, testimonials, reviews, and editorials should be excluded from the research design, as they are derived from secondary sources and would render this review tertiary. Additionally, non-research literature ought to be omitted.

2) The study's inclusion criteria are as follows: articles must be composed in English, published from 2019 to 2025, and grounded in original research that utilizes empirical or theoretical data. Furthermore, publication in academic journals, conference proceedings, or book/book sections is required for the studies.

D. Analysis Include Article

The results presented in this literature review have been obtained by extracting pertinent data from the articles. The results synthesized are based on the knowledge of the reviewers as well as the quality and substance of the available literature [20]. All results, excluding those presented in Section 3.3, were obtained through data abstraction. The NVIVO software was utilized in Section 3.3 to determine the most frequently occurring terms from a compiled text that included all analytical sections of each article in the review.

This study conducts a literature review using the SETA Systematic Evaluation of Technical Articles for SMEs organizations, utilizing the NVivo software query tool. NVivo is a software application designed specifically for executing qualitative and mixed-methods research. It is used extensively in the analysis of disorganized data. Formulate a Text Search Query to identify specific words or phrases. This software empowers users to conduct word usage, context, and meaning analyses, identify prevalent concepts or subjects within documents, and classify words or phrases automatically. The system can employ well-established synonyms to detect exact parallels or comparable terms. Furthermore, it provides advanced features, including collocation search, fuzzy search, wildcard search, and Boolean operators. Upon completing the Text Search Query, articles containing critical information will be extracted, and the primary topics will be identified.

IV. Result

The present investigation produced a total of 150 data entries. Before conducting any survey, it is critical to eradicate any occurrences of redundancy. After removing duplicate entries, the residual records amounted to 75. Screening is performed in the initial analysis phase using NVivo, employing the combination of the terms "SETA + SME + IMPORTANT + CHALLENGE." Consequently, 57 records were identified as being extraneous for this evaluation. The number of articles included in the second screening was 18, as determined by combining the criteria "SETA + SME + CHALLENGE." The papers excluded from the exhaustive text analysis lacked empirical data and were not pertinent to the subject under investigation. As a result, 57 articles remained for examination. As modified from [14], [21], Fig. 1 illustrates the screening procedure. The remaining 57 articles were then subjected to a thorough text analysis to extract relevant data and insights related to the research topic. The screening procedure outlined in Fig. 1 helped to streamline the selection process and ensure that only the most pertinent articles were included in the study. By focusing on articles that met the "SETA + SME + CHALLENGE" criteria, the researchers gathered valuable information that contributed to the overall research findings. The systematic approach to screening and analyzing articles helped maintain the study's rigor and validity.

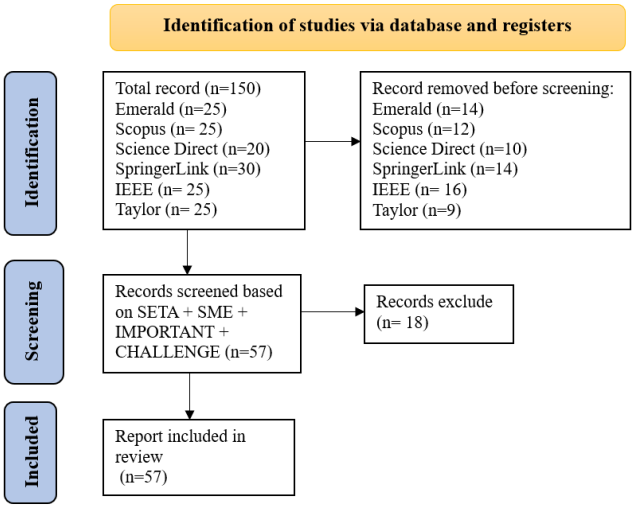


Fig. 1. The SLR screening process.

A. Finding

The field of Cybersecurity Programmes in SMEs is vast and encompasses numerous perspectives. As a result, the NVivo analytic software was employed to examine the fundamental components of cybersecurity management. Examining these fifty articles entailed the implementation of NVivo inquiry methodologies, such as Word Frequency and Text Search. As a result, these papers encompass two fundamental perspectives regarding the Cybersecurity SETA Programmes of SMEs: the importance of the SETA program and the challenges it presents. The two main perspectives discussed in this study are Cybersecurity Programmes in Small and Medium Enterprises (SMEs). This section shall explore the categories above,

commencing with the variety most frequently alluded to in the literature.

The primary aim is to utilize NVivo to identify and classify each phrase occurrence. Coding involves the methodical categorization of data to recognize recurring patterns or themes. The occurrences of the phrases "SETA," "SME," and "IMPORTANT" are identified and encoded for subsequent analysis in this instance. The result visualization in Table I provides a methodical summary of the coded phrases. This visual representation clearly explains the frequency and distribution of these key phrases within the dataset. By analyzing the patterns and themes that emerge from the coded data, researchers can gain valuable insights into the underlying concepts and trends present in the text. Overall, using NVivo for coding and analysis proves to be an effective tool in uncovering meaningful information from the data.

TABLE I. TOTAL REFERENCES BASED ON TERM

Title	Year	References based on the term
Challenges and strategies of contemporary cybersecurity awareness training in Swedish SMEs: A qualitative study [56]	2025	40
Implementation of methods to raise employees' cybersecurity awareness in small businesses [55]	2025	30
Harnessing the Right Talent for SETA Programs: Cybersecurity Roles and Competencies that Make a Difference [57]	2024	27
Employee Cybersecurity Awareness Training Programs Customized for SME Contexts to Reduce Human-Error-Related Security Incidents [58]	2024	61
An Exploratory Investigation into Sustaining Cybersecurity Protection Through the Implementation of SETA [59]	2024	32
Analysis of Information Security in a Corporate Environment – a Human Perspective [60]	2024	12
Critical success factors for Security Education, Training and Awareness (SETA) programme effectiveness [23]	2024	45
The critical success factors for Security Education, Training and Awareness (SETA) Programmes [22]	2023	50
Security culture and security education, training, and awareness (SETA) influence information security management [54]	2023	30
The Perspective of Small and Medium Enterprises SME's) and their relationship with the Government in overcoming Cybersecurity Challenges and barriers in Wales [25]	2023	43
Cyber risk assessment in small and medium-sized enterprises: A multilevel decision-making approach for small e-tailers [34]	2023	20
The effect of IT security issues on the implementation of Industry 4.0 in SMEs: Barriers and challenges [16]	2023	16
Survey and Lessons Learned on Raising SME Awareness about Cybersecurity [37]	2023	14
A Framework for The Planning and Management of Cybersecurity Projects [53] in Small and Medium-Sized Enterprises [39]	2023	19
Risk-Management Framework and Information-Security Systems for Small and Medium Enterprises (SMEs): A Meta-Analysis Approach [40]	2023	13
The Impact of Industrial Revolution 4.0 and the Future of the Workforce: A Study on Malaysian IT Professionals [45]	2023	10
A Quest for Research and Knowledge Gaps in Cybersecurity Awareness for Small and Medium-Sized Enterprises [47]	2023	18
Cyber Risk Assessment and Optimization: A Small Business Case Study [48]	2023	4
Cascading information on best practice: Cyber security risk management in UK micro and small businesses and the role of IT companies [49]	2023	9
Exploration of the Impact of Cybersecurity Awareness on Small and Medium Enterprises (SMEs) in Wales Using Intelligent Software to Combat Cybercrime [24]	2022	40
Surround yourself with your betters: Recommendations for adopting Industry 4.0 technologies in SMEs [30]	2022	24
Cybersecurity for Small and Medium-sized Enterprises (SMEs) [36]	2022	5
Strategic Cybersecurity Risk Management Practices for Information in Small and Medium Enterprises [41]	2022	26
A Survey on the Cyber Security of Small-to-Medium Businesses: Challenges, Research Focus and Recommendations [50]	2022	6
Cyber-Security culture Towards Digital Marketing Communications among Small and Medium-Sized (SME) Entrepreneurs [27]	2021	8
A Threat-Based Cybersecurity Risk Assessment Approach Addressing SME Needs [28]	2021	35
Cyber risk management in SMEs: Insights from industry surveys [35]	2021	16
Cybersecurity for SMEs: Introducing the Human Element into Socio-technical Cybersecurity Risk Assessment [38]	2021	7
The Role of Trust in the Digital Interactive Model for SME Speed Internationalization [42]	2021	11
A Cybersecurity Assessment Model for Small and Medium-Sized Enterprises [44]	2021	7
Information Security and Cybersecurity Management: A Case Study with SMEs in Portugal [46]	2021	15
A Large-Scale Interview Study on Information Security in and Attacks against Small and Medium-sized Enterprises [51]	2021	1
The Need for Information Security Management for SMEs [26]	2020	18
Factors Affecting SME Owners in Adopting ICT in Business Using Thematic Analysis [29]	2020	23
Cybersecurity Risk Management in Small and Medium-Sized Enterprises: A Systematic Review of Recent Evidence [32]	2020	9
Organizational cybersecurity readiness in the ICT sector: A Quanti-Qualitative assessment [33]	2020	9
I am not usually the one who handles it: Exploring the disconnect between corporate security policies and actual security practices in SMEs [43]	2020	7
A Review of the Current Cyber Hygiene in Small and Medium-sized Businesses [52]	2020	1
Developing cybersecurity education and awareness programs for small- and medium-sized enterprises (SMEs) [31]	2019	23

1) Importance of SETA SME: 57 journal articles out of 75 submitted in NVivo contain references and content pertinent to the "IMPORTANT SETA SME." Using the acronym "SETA SME" to represent a unique subject or idea is highly probable in 39 of the 75 scholarly articles. NVivo facilitates the

classification and analysis of content by subjects or keywords. It appears to have identified publications relevant to the subject matter or engaged in discussions regarding the importance of "SETA SME." In academic articles, the reference count represents the number of citations or sources incorporated within the respective work. The observed diversity among these articles may indicate discrepancies in the depth of investigation, the complexity of the topics explored, or the extent to which each article depends on external sources to support its claims. The findings underscore the significance of SETA within the SME sector, as evidenced by the substantial body of literature devoted to this subject. This body of literature highlights the SETA's impact on small and medium enterprises, showcasing its relevance and importance in driving growth and innovation in this sector. By delving into the various perspectives and research findings presented in these articles, a more comprehensive understanding of SETA SME can be achieved, ultimately leading to informed decision-making and strategic planning for businesses operating within this realm. As such, continued exploration and analysis of this topic will undoubtedly contribute to SMEs' ongoing development and success worldwide.

2) Challenge in SETA SME: The central focus was the convergence of SETA and SMEs, which represented a targeted investigation into the challenges faced by SMEs in the context of security training and education awareness initiatives. Once the criteria were applied, NVivo ascertained that out of the 75 publications examined, 15 specifically addressed challenges associated with SETA in SMEs. As shown in Table II, this phase presumably involved the application of automatic coding or manual annotation to identify instances in which challenges were explicitly addressed in the selected publications. Further analysis and scrutiny may be conducted using the selected subset of 19 articles. The extensive array of challenges identified in the literature, including but not limited to limited resources, regulatory obstacles, and other distinctive concerns, may necessitate the application of theme analysis.

TABLE II. CHALLENGE OF THE SETA IN SME

Challenges	Term based on the article
Cybersecurity threats	40
IT risk management	38
Gaps in established frameworks	36
Cultural issues	35
Technical hurdles	30
Limited resources and expertise in SMEs	24
Awareness and knowledge gap	20
Lack of support and commitment	17
Sustaining digital projects across various domains	15
Limited Rapid ICT development poses	10

V. DISCUSSION

This study contributes to the field by presenting a taxonomy of challenges SMEs face in implementing Security Education, Training, and Awareness (SETA) programs, derived from a systematic review of 57 empirical studies. These challenges are categorized into six domains: financial constraints, human resource limitations, management support, cultural resistance, technical infrastructure gaps, and legal/data protection concerns. For instance, financial and personnel shortages remain persistent across global contexts, limiting SMEs' ability to adopt sustained cybersecurity practices [15], [60].

Additionally, this review reveals that SMEs often lack clear frameworks for assessing cybersecurity risk and executing structured training protocols, leading to inconsistent adoption of best practices such as secure password management and software updates [6], [58].

Based on thematic coding using NVivo, we also provide empirical insights into common barriers SMEs face during SETA implementation, many of which stem from poor management buy-in and low employee engagement. This reinforces previous findings that organizational culture significantly impacts the success of cybersecurity training programs [17], [57].

Crucially, the study proposes practical recommendations tailored to SME environments. These include prioritizing modular SETA programs to reduce costs, integrating role-specific cybersecurity training, leveraging government-supported platforms like CyberSAFE Malaysia, and establishing clearer metrics for evaluating training effectiveness. As SMEs remain underrepresented in cybersecurity research, our findings also identify a research gap in longitudinal evaluations of SETA effectiveness, an area where future studies should focus.

TABLE III. TAXONOMY OF SETA IMPLEMENTATION CHALLENGES IN SMES AND PRACTICAL RECOMMENDATIONS

Challenge Category	Key Issues Identified	Practical Recommendations
Financial Constraints	Limited budgets for cybersecurity tools, training, or staff	Use modular or free SETA tools (e.g., CyberSAFE Malaysia); apply for national funding grants
Human Resource Limitations	Lack of skilled cybersecurity staff	Upskill existing staff; partner with external consultants or universities
Management Support	Low executive commitment; no budget or time allocation	Integrate cybersecurity goals into business KPIs; awareness sessions for decision-makers
Cultural Resistance	Employee apathy, change resistance	Gamify training; link cyber hygiene to personal/workplace safety
Technical Infrastructure Gaps	Outdated systems, lack of firewalls, insecure networks	Adopt cloud-based SME-friendly cybersecurity solutions
Legal/Data Protection Concerns	Lack of awareness of local/international data laws	Include regulatory compliance modules in SETA training programs

Table III summarizes a taxonomy of key challenges SMEs face in implementing Security Education, Training, and Awareness (SETA) programs, along with practical recommendations to overcome each challenge. The challenges

are grouped into six categories: financial constraints, limited human resources, lack of management support, cultural resistance, technical infrastructure gaps, and legal/data protection issues. For each category, the table outlines actionable solutions such as adopting low-cost training tools, partnering with external experts, integrating cybersecurity into business KPIs, engaging employees through personalized training, leveraging cloud-based security solutions, and including legal compliance in SETA content. This structured framework serves as a practical guide for SMEs and policymakers to strengthen cybersecurity readiness in resource-constrained environments.

This review not only confirms the importance of SETA in SMEs but also maps out the landscape of specific barriers, provides a taxonomy for future research, and offers actionable recommendations for industry practitioners and policymakers.

VI. Conclusion

This study conducted a systematic literature review (SLR) to explore the importance and challenges of implementing Security Education, Training, and Awareness (SETA) programs in Small and Medium Enterprises (SMEs), particularly within the Malaysian context. Drawing from 57 relevant academic sources, the study developed a taxonomy of six major challenge areas: financial, human resource, management, cultural, technical, and legal, providing both academic clarity and practical recommendations for improving cybersecurity practices in SMEs.

One of the central contributions of this research lies in synthesizing fragmented insights into a cohesive framework, which can guide policymakers and SME decision-makers in designing more tailored and effective SETA initiatives. The findings underscore the need for modular, cost-effective, and context-sensitive training, especially given SMEs' limitations in budget, staffing, and technical infrastructure.

However, it is important to acknowledge that this study remains descriptive in nature, as it is based solely on secondary data from published literature. The study does not include empirical validation with actual SME stakeholders, such as through surveys, interviews, or field studies. This is a notable limitation, as the practical relevance and applicability of the proposed taxonomy and recommendations could vary when applied to real-world SME contexts.

VII. Future Work

To address this gap, future research should incorporate empirical methods to validate the SLR findings. Mixed-method approaches—such as interviews with SME managers, surveys assessing SETA adoption, or pilot testing of tailored training models—can help confirm the practical relevance of the challenges and recommendations identified. Collaborations with local SMEs, government agencies, and cybersecurity training providers would also enable researchers to co-create field-tested frameworks that bridge the divide between academic theory and practical implementation.

By extending this research into applied settings, scholars and practitioners can more effectively support the cybersecurity resilience of SMEs, ensuring that SETA programs are not only theoretically sound but also operationally viable and sustainable in real-world business environments.

Acknowledgment

This work was supported by the Yayasan Canselor UNITEN Community Energy Grant. The authors thank Yayasan Canselor UNITEN for funding that enabled the systematic review and analysis presented in this paper. The support was instrumental in advancing the study of cybersecurity awareness and SETA implementation in the SME sector. The authors also acknowledge Universiti Tenaga Nasional (UNITEN) for institutional support throughout the research process.

References

[1] EC-Council, "Cybersecurity awareness in Malaysia."

[2] International Trade Administration, "MALAYSIA CYBERSECURITY."

[3] Cyber Security Malaysia, "Power of ISMS: Driving Value & Growth for SMEs in Malaysia," 2021.

[4] Ethan Yeo, "Invest in Cyber Security, SMEs Urged," The Sun, Sep. 04, 2023.

[5] K. Khando, S. Gao, S. M. Islam, and A. Salman, "Enhancing employees' information security awareness in private and public organizations: A systematic literature review," Comput Secur, vol. 106, p. 102267, Jul. 2021, doi: 10.1016/j.cose.2021.102267.

[6] Aiza Azreen Ahmad, "Building Trusted, Secure and Ethical Digital Environment for Malaysian SMEs," The Edge Malaysia, Jul. 16, 2021.

[7] L.-W. Wong, V.-H. Lee, G. W.-H. Tan, K.-B. Ooi, and A. Sohal, "The role of cybersecurity and policy awareness in shifting employee compliance attitudes: Building supply chain capabilities," Int J Inf Manage, vol. 66, p. 102520, Oct. 2022, doi: 10.1016/j.ijinfomgt.2022.102520.

[8] S. Hu, C. Hsu, and Z. Zhou, "The impact of SETA event attributes on employees' security-related Intentions: An event system theory perspective," Comput Secur, vol. 109, p. 102404, Oct. 2021, doi: 10.1016/j.cose.2021.102404.

[9] Cyber Security Malaysia, "Program Galakan Pemerkasan Keselamatan Siber Kepada PKS."

[10] Mohd Nazer Apau and Sabariah Ahmad, "Information Security Guidelines for Small & Medium Enterprises (SMEs)," 2020.

[11] S. R. Hamidi, A. A. Aziz, S. M. Shuhidan, A. A. Aziz, and M. Mokhsin, "SMEs maturity model assessment of IR4.0 digital transformation," Advances in Intelligent Systems and Computing, vol. 739, pp. 721–732, 2018, doi: 10.1007/978-981-10-8612-0_75.

[12] Chubb Insurance Malaysia Berhad, "How SMEs can protect themselves from cyber risks." Accessed: Sep. 15, 2023. [Online]. Available: https://www.chubb.com/my-en/articles/business/how-smes-can-protect-themselves-from-cyber-risks.html

[13] A. Santos-Olmo, L. Sánchez, I. Caballero, S. Camacho, and E. Fernandez-Medina, "The Importance of the Security Culture in SMEs as Regards the Correct Management of the Security of Their Assets," Future Internet, vol. 8, no. 4, p. 30, Jul. 2016, doi: 10.3390/fi8030030.

[14] Amrita V. Nair, "Accelerating Small and Medium-Sized Enterprise (SME) Digitalisation in Malaysia," 2023.

[15] Daniella Balaban, "Security Challenges of SMEs (Small to Medium Enterprises)."

[16] M. F. Arroyabe, C. F. A. Arranz, I. F. de Arroyabe, and J. C. F. de Arroyabe, "The effect of IT security issues on the implementation of Industry 4.0 in SMEs: Barriers and challenges," Technol Forecast Soc Change, vol. 199, no. December 2023, p. 123051, 2024, doi: 10.1016/j.techfore.2023.123051.

[17] S. Saeed, S. A. Suayyid, M. S. Al-Ghamdi, H. Al-Muhaisen, and A. M. Almuhaideb, "A Systematic Literature Review on Cyber Threat Intelligence for Organizational Cybersecurity Resilience," Sensors, vol. 23, no. 16, p. 7273, Aug. 2023, doi: 10.3390/s23167273.

[18] M. F. Arroyabe, C. F. A. Arranz, I. F. de Arroyabe, and J. C. F. de Arroyabe, "The effect of IT security issues on the implementation of industry 4.0 in SMEs: Barriers and challenges," Technol Forecast Soc Change, vol. 199, p. 123051, Feb. 2024, doi: 10.1016/j.techfore.2023.123051.

[19] D. Tranfield, D. Denyer, and P. Smart, "Towards a Methodology for Developing Evidence‐Informed Management Knowledge using Systematic Review," British Journal of Management, vol. 14, no. 3, pp. 207‐222, Sep. 2003, doi: 10.1111/1467-8551.00375.

[20] A. Fink, Conducting Research Literature Reviews: From the Internet to Paper., Fifth. Sage Publications, 2020.

[21] M. J. Page et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," BMJ, p. n71, Mar. 2021, doi: 10.1136/bmj.n71.

[22] A. Alyami, D. Sammon, K. Neville, and C. Mahony, "Critical success factors for Security Education, Training and Awareness (SETA) program effectiveness: an empirical comparison of practitioner perspectives," Information & Computer Security, Aug. 2023, doi: 10.1108/ICS-08-2022-0133.

[23] A. Alyami, D. Sammon, K. Neville, and C. Mahony, "The critical success factors for Security Education, Training and Awareness (SETA) program effectiveness: a lifecycle model," Information Technology and People, vol. 36, no. 8, pp. 94–125, 2023, doi: 10.1108/ITP-07-2022-0515.

[24] N. Rawindaran, A. Jayal, and E. Prakash, "Exploration of the Impact of Cybersecurity Awareness on Small and Medium Enterprises (SMEs) in Wales Using Intelligent Software to Combat Cybercrime," Computers, vol. 11, no. 12, 2022, doi: 10.3390/computers11120174.

[25] N. Rawindaran, A. Jayal, E. Prakash, and C. Hewage, "Perspective of small and medium enterprises (SME's) and their relationship with government in overcoming cybersecurity challenges and barriers in Wales," International Journal of Information Management Data Insights, vol. 3, no. 2, p. 100191, 2023, doi: 10.1016/j.jjimei.2023.100191.

[26] M. I. Khan, S. Tanwar, and A. Rana, "The need for information security management for SMEs," Proceedings of the 2020 9th International Conference on System Modeling and Advancement in Research Trends, SMART 2020, pp. 328–332, 2020, doi: 10.1109/SMART50582.2020.9337108.

[27] A. H. Adleena Huzaizi, S. N. A. Ahmad Tajuddin, K. A. Bahari, K. A. Manan, and N. N. Abd Mubin, "Cyber-Security Culture towards Digital Marketing Communications among Small and Medium-Sized (SME) Entrepreneurs," Asian Culture and History, vol. 13, no. 2, p. 20, 2021, doi: 10.5539/ach.v13n2p20.

[28] M. Van Haastrecht, I. Sarhan, A. Shojaifar, L. Baumgartner, W. Mallouli, and M. Spruit, "A Threat-Based Cybersecurity Risk Assessment Approach Addressing SME Needs," ACM International Conference Proceeding Series, 2021, doi: 10.1145/3465481.3469199.

[29] A. N. A. Rozmi, P. N. E. Nohuddin, A. R. A. Hadi, M. I. A. Bakar, and A. I. Nordin, "Factors affecting SME owners in adopting ICT in business using thematic analysis," International Journal of Advanced Computer Science and Applications, vol. 11, no. 7, pp. 208–218, 2020, doi: 10.14569/IJACSA.2020.0110727.

[30] O. F. Grooss, M. Presser, and T. Tambo, "Surround yourself with your betters: Recommendations for adopting Industry 4.0 technologies in SMEs," Digital Business, vol. 2, no. 2, p. 100046, 2022, doi: 10.1016/j.digbus.2022.100046.

[31] M. Bada and J. R. C. Nurse, "Developing cybersecurity education and awareness programmes for small- and medium-sized enterprises (SMEs)," Information and Computer Security, vol. 27, no. 3, pp. 393–410, 2019, doi: 10.1108/ICS-07-2018-0080.

[32] Awareness, Data Analytics and Assessment, Cyber SA 2020, 2020, doi: 10.1109/CyberSA49311.2020.9139638.

[33] M. Neri, F. Niccolini, and L. Martino, "Organizational cybersecurity readiness in the ICT sector: a quanti-qualitative assessment," Information and Computer Security, 2023, doi: 10.1108/ICS-05-2023-0084.

[34] A. Sukumar, H. A. Mahdiraji, and V. Jafari-Sadeghi, "Cyber risk assessment in small and medium-sized enterprises: A multilevel decision-making approach for small e-tailers," Risk Analysis, vol. 43, no. 10, pp. 2082–2098, 2023, doi: 10.1111/risa 14092.

[35] F. Hoppe, N. Gatzert, and P. Gruner, "Cyber risk management in SMEs: insights from industry surveys," Journal of Risk Finance, vol. 22, no. 3–4, pp. 240–260, 2021, doi: 10.1108/JRF-02-2020-0024.

[36] V. Arya, "Cybersecurity for Small and Medium-sized Enterprises (SMEs)," vol. 5, pp. 7–10, 2022.

[37] [37] C. Ponsard, J. Grandclaudon, and S. Bal, "Survey and Lessons Learned on Raising SME Awareness about Cybersecurity," International Conference on Information Systems Security and Privacy, no. Icissp, pp. 558–563, 2019, doi: 10.5220/0007574305580563.

[38] C. Boletsis, R. Halvorsrud, J. B. Pickering, S. Phillips, and M. Surridge, "Cybersecurity for SMEs: Introducing the human element into socio-technical cybersecurity risk assessment," VISIGRAPP 2021 - Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, vol. 3, no. Visigrapp, pp. 266–274, 2021, doi: 10.5220/0010332902660274.

[39] B. Von Skarczinski et al., "A framework for the planning and management of cybersecurity projects in small and medium-sized enterprises," Information and Computer Security, vol. 3, no. 2, pp. 10–37, 2023, doi: 10.1145/3465481.3469199.

[40] K. AL-Dosari and N. Fetais, "Risk-Management Framework and Information-Security Systems for Small and Medium Enterprises (SMEs): A Meta-Analysis Approach," Electronics (Switzerland), vol. 12, no. 17, 2023, doi: 10.3390/electronics12173629.

[41] C. Ashley and M. Preiksaitis, "Strategic Cybersecurity Risk Management Practices for Information in Small and Medium Enterprises," Business Management Research and Applications: A Cross-Disciplinary Journal, vol. 1, no. 2, pp. 109–157, 2022, [Online]. Available: https://bmrajournal.columbiasouthern.edu/index.php/bmra/article/view/3421

[42] A. Mohamad, A. Mohd Rizal, H. Khalid, and T. H. Char Fei, "The Role of Trust in the Digital Interactive Model for SME Speed Internationalisation," International Conference on Research and Innovation in Information Systems, ICRIIS, pp. 1–6, 2021, doi: 10.1109/ICRIIS53035.2021.9617095.

[43] M. Sadok, S. Alter, and P. Bednar, "It is not my job: exploring the disconnect between corporate security policies and actual security practices in SMEs," Information and Computer Security, vol. 28, no. 3, pp. 467–483, 2020, doi: 10.1108/ICS-01-2019-0010.

[44] A. Emer, M. Unterhofer, and E. Rauch, "A Cybersecurity Assessment Model for Small and Medium-Sized Enterprises," IEEE Engineering Management Review, vol. 49, no. 2, pp. 98–109, 2021, doi: 10.1109/EMR.2021.3078077.

[45] M. H. Husin, N. F. Ibrahim, N. A. Abdullah, S. M. Syed-Mohamad, N. H. Samsudin, and L. Tan, "The Impact of Industrial Revolution 4.0 and the Future of the Workforce: A Study on Malaysian IT Professionals," Soc Sci Comput Rev, vol. 41, no. 5, pp. 1671–1690, 2023, doi: 10.1177/08944393221117268.

[46] Mário Antunes, Marisa Maximiano, Ricardo Gomes, and Daniel Pinto, "Information Security and Cybersecurity Management: A Case Study with SMEs in Portugal," Journal of CyberSecurity and Privacy, pp. 219–238, 2021.

[47] S. Chaudhary, V. Gkioulos, and S. Katsikas, "A quest for research and knowledge gaps in cybersecurity awareness for small and medium-sized enterprises," Comput Sci Rev, vol. 50, no. 830929, p. 100592, 2023, doi: 10.1016/j.cosrev.2023.100592.

[48] M. Tsiodra, S. Panda, M. Chronopoulos, and E. Panaousis, "Cyber Risk Assessment and Optimization: A Small Business Case Study," IEEE Access, vol. 11, no. April, pp. 44467–44481, 2023, doi: 10.1109/ACCESS.2023.3272670.

[49] A. Cartwright, E. Cartwright, and E. S. Edun, "Cascading information on best practice: Cyber security risk management in UK micro and small businesses and the role of IT companies," Comput Secur, vol. 131, 2023, doi: 10.1016/j.cose.2023.103288.

[50] A. Chidukwani, S. Zander, and P. Koutsakis, "A Survey on the Cyber Security of Small-to-Medium Businesses: Challenges, Research Focus, and Recommendations," IEEE Access, vol. 10, no. August, pp. 85701–85719, 2022, doi 10.1109/ACCESS.2022.3197899.

[51] N. Huaman et al., A Large-Scale Interview Study on Information Security in and Attacks against Small and Medium-sized Enterprises. [Online].

Available: https://www.usenix.org/conference/usenixsecurity21/presentation/huaman

[52] T. Ncubukezi, L. Mwansa, and F. Rocaries, "A Review of the Current Cyber Hygiene in Small and Medium-sized Businesses," 2020 15th International Conference for Internet Technology and Secured Transactions, ICITST 2020, 2020, doi: 10.23919/ICITST51030.2020.9351339.

[53] M. Figueredo Franco, F. Martins Lacerda, and B. Stiller, "A framework for the planning and management of cybersecurity projects in small and medium-sized enterprises," Revista de Gestão e Projetos, vol. 13, no. 3, pp. 10–37, 2022, doi: 10.5585/gep.v13i3.23083.

[54] Heyasat, H., Mubarak, S., Evans, N. Security Culture and Security Education, Training and Awareness (SETA) Influencing Information Security Management. In: Daimi, K., Al Sadoon, A. (eds) Proceedings of the Second International Conference on Innovations in Computing Research (ICR'23). Lecture Notes in Networks and Systems, 2023. vol 721. Springer, Cham. https://doi.org/10.1007/978-3-031-35308-6_28

[55] Jonas Schweizer, "Implementation of methods to raise employees' cybersecurity awareness in small businesses."Information Systems, Social aspects. 2025. , p. 86, iv

[56] Tabassum, Jinia. "Challenges and strategies of contemporary cybersecurity awareness training in Swedish SMEs: A qualitative study."Information Systems, Social aspects. 2025. , p. 86, iv

[57] Charalambous, A., & Stavrou, E. Harnessing the Right Talent for SETA Programs: Cybersecurity Roles and Competencies that Make a Difference. 2024. , 130-144. https://doi.org/10.1007/978-3-031-72563-0_10.

[58] Ugbebor, F., Aina, O., Abass, M., & Kushanu, D. Employee Cybersecurity Awareness Training Programs Customized for SME Contexts to Reduce Human-Error Related Security Incidents. Journal of Knowledge Learning and Science Technology ISSN: 2024. 2959-6386 (online). https://doi.org/10.60087/jklst.vol3.n3.p382-409.

[59] A., Huda, M., Brayyich, M., Haroon, N., Khan, A., & Verma, R. An Exploratory Investigation into Sustaining Cybersecurity Protection Through the Implementation of SETA. 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), 292-296. https://doi.org/10.1109/ETNCC63262.2024.10767573.

[60] Tick, A., & Szabó-Harka, N. Analysis of Information Security in a Corporate Environment – a Human Perspective. 2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI), 2-24. 000469-000474. https://doi.org/10.1109/SAMI60510.2024.10432889.

Hybrid Fuzzy–PPO Control for Precision UAV Spraying

Ahmad B. Alkhodre[1], Adnan Ahmed Abi Sen[2], Yazed Alsaawy[3], Nour Mahmoud Bahbouh[4], Mohamed Benaida[5]
Faculty of Computer and Information Systems, Islamic University of Madinah, Saudi Arabia [1, 3, 5]
Dr. Hussein ElSayyed Research Center-Department of Graduate Studies & Scientific Research, University of Prince Mugrin, Madinah 42241, Saudi Arabia[2]
Department of Information and Communication Sciences, University of Granada, 18071 Granada, Spain[4]

***Abstract*—Precision agriculture increasingly relies on autonomous UAVs for tasks, such as crop monitoring and targeted pesticide spraying. However, maintaining stable flight and precise spray delivery under varying payloads and wind disturbances remains challenging. This paper proposes a hybrid control architecture that combines interpretable Mamdani fuzzy logic controllers with a deep reinforcement learning (DRL) agent (Proximal Policy Optimization, PPO). The fuzzy controllers encode expert-crafted rules for baseline altitude and attitude stabilization, while the PPO agent adaptively adjusts setpoints to optimize spray coverage and energy efficiency. We train the agent in a realistic PyBullet simulator with dynamic payload and wind conditions. In simulated precision-spraying trials, our hybrid controller outperformed both a conventional PID-based controller and a pure PPO controller. Specifically, it achieved roughly 2–3× faster disturbance rejection, near-zero overshoot, and ~30% faster settling than the baselines, resulting in more uniform coverage and reduced pesticide use. These results demonstrate that fusing fuzzy logic with deep PPO yields a UAV spray controller that is both high-performance and robust for precision agriculture applications.**

***Keywords*—UAVs; precision agriculture; UAV spraying; fuzzy logic control; reinforcement learning; Proximal Policy Optimization (PPO); hybrid control**

I. Introduction

Unmanned Aerial Vehicles (UAVs) have become vital tools in precision agriculture for tasks such as crop monitoring, health assessment, and targeted spraying. Compared to ground-based systems or manned spraying, UAVs can achieve more uniform coverage and reduced chemical usage [1, 2]. However, realizing fully autonomous spraying requires robust flight control: the UAV must maintain accurate altitude and attitude while compensating for external disturbances (e.g., wind gusts) and internal changes (e.g. decreasing payload mass as pesticide is dispensed) [3]. Traditional UAV controllers use cascaded proportional–integral–derivative (PID) loops for altitude and attitude stabilization. While PID controllers perform well under nominal conditions, a fixed-gain PID tuned for one weight or calm wind conditions can exhibit large errors, overshoot, or oscillations if the payload changes or a sudden gust occurs [3]. These limitations motivate more adaptive, nonlinear control strategies.

Nonlinearities without an explicit plant model. Mamdani-type fuzzy controllers have been successfully applied to quadcopter attitude and altitude control. Fuzzy controllers often achieve faster response and minimal steady-state error in certain scenarios, thanks to their ability to encode heuristic rules (e.g., "if altitude error is positive and increasing, then apply strong downward thrust") [16]. Similarly, researchers used Takagi–Sugeno fuzzy models to reduce controller complexity and still achieve accurate tracking [16]. Overall, fuzzy logic offers interpretability and robustness, which is valuable for safety-critical UAV tasks. However, pure fuzzy controllers do not learn or adapt online, and designing membership functions and rule bases typically requires manual tuning.

Deep Reinforcement Learning (DRL) showed promise for UAV control by learning policies through trial-and-error in simulation [3, 4]. Modern policy-gradient algorithms such as PPO [13] can train controllers that adapt to complex dynamics. Other works combined PID with DRL to achieve fast stability: Wu et al. used a corrective-feedback DRL scheme for UAV landing and mixed a PID baseline with learned adjustments, and Ma et al. applied deep RL to reject wind disturbances, demonstrating improved stability in level-5 wind conditions [5, 6]. Nevertheless, DRL requires extensive data and yields black-box policies, raising safety and interpretability concerns.

To leverage the strengths of both approaches, hybrid fuzzy–RL architectures have been proposed. A common scheme is to run a fuzzy (or PID) controller in the inner loop for basic stability, while an RL agent provides high-level setpoints or bias corrections. For example, [14] introduced a Fuzzy-PPO controller for a guided vehicle, where the PPO agent augmented a fuzzy system and reduced tracking path errors compared to a fuzzy-based controller (average error from 0.05 m to 0.02 m). These studies suggest that integrating fuzzy rules can accelerate learning and improve safety; however, to our knowledge, none have specifically addressed precision UAV spraying with a Mamdani fuzzy–PPO scheme.

In this paper, we apply the hybrid fuzzy–RL concept to precision UAV spraying. We design a Mamdani fuzzy control system for altitude, pitch, and roll stabilization, and integrate it with an adaptive learning (PPO) agent. The fuzzy controllers provide baseline stability via intuitive rules, while the agent adjusts setpoints to optimize spray accuracy and energy. We train and evaluate our architecture in a detailed PyBullet simulation with wind and payload variation [15, 35].

In this work, we introduce a novel hybrid Mamdani fuzzy–PPO controller tailored for a quadrotor pesticide sprayer. Our contributions are:

- Hybrid control design: We develop Mamdani fuzzy controllers for altitude, pitch, and roll stabilization and integrate them with a PPO agent. The fuzzy layer ensures immediate stability through intuitive rules, while the PPO agent supplies adaptive setpoint adjustments. This hierarchical design combines expert knowledge and learning-based adaptability in a unified framework.
- Reward shaping for spray tasks: We introduce reward functions that balance tracking precision with control effort and spray uniformity. By penalizing overshoot and incentivizing steady flight, the shaped rewards accelerate RL training and lead to efficient spray trajectories.
- Comprehensive evaluation: We conduct extensive simulations comparing our hybrid controller against strong baselines: a well-tuned PID controller, a standalone PPO agent, and (qualitatively) advanced adaptive baselines. The hybrid controller consistently outperforms all alternatives: it rejects disturbances roughly 2–3× faster and settles ~30% quicker, with minimal overshoot and lower energy use (see Table IV).
- Operational benefits: We quantify practical gains in spraying scenarios: the hybrid system achieves more uniform chemical coverage, reduces energy consumption, and maintains stability under variable load and wind. These improvements translate into cost savings and sustainability in agricultural operations (less wasted pesticide, longer flight time).

Overall, our results demonstrate that combining an interpretable fuzzy baseline with deep PPO yields a UAV controller that is both accurate and robust for precision agriculture spraying [46].

The rest of this paper is organized as follows: Section II presents the literature review, Section III details our methodology (system design, fuzzy logic control, PPO integration), Section IV presents experimental results and comparisons, and Section V concludes with discussion of implications and future work.

II. Literature Review

We classified the previous studies related to our subject into five categories, namely:

A. UAVs in Precision Agriculture

UAV platforms have transformed agricultural practices by providing flexible, on-demand aerial applications. Recent surveys highlight the wide range of precision farming uses for drones, including crop monitoring, variable-rate application, and targeted spraying [2]. Delavarpour et al. (2023) review 213 sources on UAV sprayers and emphasize that UAVs can deliver chemicals more precisely than manned aircraft, though autonomy and control remain key gaps [2]. RL has also been integrated with perception for precision-ag UAVs [39], and deep RL has been applied to agricultural path planning [8, 26]. Spray accuracy is particularly sensitive to wind: field studies report that even moderate winds can carry droplets off-target, reducing efficacy [19]. These agricultural factors confirm the need for accurate flight control. To maximize spray (coverage) uniformity, the UAV must maintain steady height and heading so the spray path aligns with the crop rows, even as payload mass varies. Thus, well-regulated flight attitude improves coverage, saves energy, and ensures safety [9, 10]. In summary, the precision agriculture domain sets hard demands: controllers must handle nonlinear dynamics, time-varying payloads, and external disturbances, all while optimizing coverage efficiency.

B. Classical Control: PID and Variants

Traditionally, UAV flight control systems employ cascaded PID loops for altitude and attitude stabilization. These controllers are simple and effective in nominal conditions [21, 22, 23]. However, PID has notable limitations. First, it assumes a linear response around an operating point. Quadrotor dynamics are nonlinear and under-actuated (four inputs for six degrees of freedom), leading to cross-axis couplings that a separate PID per axis cannot fully address [16]. Second, PID parameters are usually tuned for a particular mass and airframe configuration. In practice, as payload weight decreases (when spray is used up) or when external forces change, the fixed gains no longer yield optimal performance. As Koch et al. [3] remark, under unknown wind or payload changes, "a PID controller can be far from optimal." This motivates exploration of nonlinear and learning-based alternatives [45, 32, 38, 33].

C. Fuzzy Logic Control for UAVs

Fuzzy logic controllers (FLCs) have been widely studied for UAV stability control because they can capture heuristic expert knowledge and handle nonlinearities without an explicit model [16]. A fuzzy controller defines a set of IF–THEN rules. In a Mamdani FLC, the output of each rule is a fuzzy set, which is defuzzified to produce a crisp control action. The rule base can be constructed manually based on physical insight or tuned via data. For quadrotor control, Mamdani FLCs have been applied to altitude and attitude loops. Studies show that a fuzzy PD controller can outperform a conventional PID (faster rise time and smaller overshoot) [12, 16, 40, 43]. Moreover, some researchers proposed adaptive fuzzy schemes to handle disturbances [25, 28]. For example, Coza et al. developed an adaptive-fuzzy controller that adjusts membership centers online to account for wind disturbances, and maintain stability without chattering [30]. Robust and adaptive fuzzy trajectory tracking further mitigates disturbances [44], including advanced T–S fuzzy designs with LMI tuning [41].

Overall, fuzzy controllers offer robustness to uncertainty and do not require accurate system identification, and can handle variable load effects and average wind by adjusting control. However, they do not inherently adapt online; designing effective membership functions and rule bases still requires expert knowledge or formal methods [16, 37]. Additional hovering control studies corroborate these gains [42].

D. Deep Reinforcement Learning for UAV Control

Deep reinforcement learning (DRL) has emerged as a powerful tool for UAV control by leveraging neural networks to learn complex policies [36]. Early works applied DRL to high-level tasks (e.g. navigation), but more recent research has moved

RL into the inner loop. Koch et al. showed that PPO produced smoother control signals and lower roll/yaw error than PID in an inner-loop attitude task, supporting its suitability for high-precision control [47]. Subsequent studies applied DRL to various UAV tasks. Wu et al. (2022) [5] used a DDPG-based agent with corrective feedback to land a UAV on a moving platform; by blending a PID baseline with learned adjustments, they achieved precise landings under dynamic conditions. Hybridizing RL with Sliding-Mode control also reduced chattering in attitude control [27]. Ma et al. (2024) trained a deep RL policy specifically to reject wind disturbances, maintaining level flight in gusts that destabilized fixed PID [6]. In precision spraying missions, hierarchical action-masked PPO improved navigation and spray efficiency [31], and conditional-action-tree RL achieved similar gains in agricultural tasks [24]. Complementarily, RL has also been used to tune classical PD gains online with successful real-flight tests [34]. Despite these advances, pure RL policies remain data-hungry and less transparent, raising safety and interpretability concerns.

E. Hybrid Fuzzy-RL Architectures

Motivated by the complementary strengths of fuzzy logic and DRL, recent research has proposed hybrid controllers. The key idea is to retain a rule-based fuzzy or classical core for stability and let a learning agent tweak setpoints or feedforward bias for performance and adaptability. One general strategy is a cascaded architecture, where the fuzzy (or PID) controller runs in the inner loop to provide safe control, and the RL agent acts as an outer-loop "bias" or feed-forward adjustment [48]. This way, the RL agent does not have to learn basic stabilization from scratch, speeding up training and improving reliability. Kuo et al. (2025) [14] developed a Fuzzy PPO (FPPO) controller for an autonomous guided vehicle (AGV) path tracking task. Xia et al. (2024) [11] propose a different hybrid: they combine a Soft Actor-Critic (SAC) DRL agent with a Fuzzy Inference System (FIS) for UAV target interception [49]. The SAC–FIS agent splits control among subsystems using fuzzy logic for attitude commands and deep learning for navigation; the fuzzy component provides "universal experiences" (prior knowledge) that reduce training time and control cost. Their results support the idea that fuzzy rules accelerate learning by narrowing the policy search.

In summary, while these efforts are promising, to our knowledge none have specifically addressed precision UAV spraying with a hybrid Mamdani fuzzy–PPO scheme. As summarized in Table I, fuzzy controllers and DRL controllers each bring complementary advantages—fuzzy logic yields robustness and interpretability, whereas DRL offers adaptability and self-tuning—while hybrid approaches attempt to integrate these benefits. Our work extends this line by focusing on a spraying UAV: we propose the first architecture that merges Mamdani fuzzy control with a PPO agent for precision agricultural spraying, and we provide a detailed simulation study demonstrating the resulting performance gains. The next sections describe our methodology and experiments.

TABLE I COMPARATIVE SUMMARY OF UAV CONTROL APPROACHES (FIXED-GAIN PID, FUZZY LOGIC CONTROL, DRL, AND HYBRID FUZZY–RL): KEY STRENGTHS, WEAKNESSES

Approach	Strengths	Weaknesses	Representative Works
Fixed-Gain PID	Simple design, proven stability	Performance degrades under payload/wind changes	[20]
Fuzzy Logic Control	Handles nonlinearity, robust to uncertainty	Requires expert rule tuning, no automatic adaptation	[25] [30]
DRL (PPO/DDPG, etc.)	Adaptive, model-free, can optimize performance	Data-hungry, opaque policies, safety concerns	[3], [5]
Hybrid Fuzzy–RL	Combines interpretability with adaptivity	Higher design complexity, few prior examples	[14], [11]

III. METHODOLOGY

In this section, we present the design of the proposed hybrid control system and explain its key components. We also discuss the mathematical foundations behind both the fuzzy logic framework and the reinforcement learning approach that drive its performance.

A. System Design

The proposed UAV control system consists of a hierarchical architecture where a high-level learning agent provides adaptive setpoints that are tracked by low-level fuzzy controllers. Fig. 1 shows a schematic of the hybrid control system, which includes the UAV plant, sensors, PPO RL agent, fuzzy inference controllers, and the environmental disturbance inputs. In this architecture, the UAVs provide sensors (e.g., IMU, altimeter, GPS, wind sensor) that provide state measurements such as altitude, roll, pitch, yaw angles, angular rates, and battery/throttle status. These sensor data are fed into two parallel control modules: 1) the RL agent (PPO policy network) and 2) the fuzzy logic controllers. The RL agent observes the state and environmental context (like current wind speed/direction and remaining payload) as its input. At each time step (e.g. 10 Hz control loop), the PPO policy computes a high-level action – in our design, this action is a setpoint or adjustment for the desired flight state. Specifically, the PPO outputs a target altitude and target pitch/roll angles (or equivalently, target horizontal velocities) that it deems optimal for the current conditions. These serve as dynamic references that can change in anticipation of or reaction to disturbances.

The PPO agent's action vector is given as:

$$a_k \in [-1,1]^4 \tag{1}$$

This is rescaled to:

$$\overline{a_k} = \frac{a_k+1}{2} \in [0,1]^4 \tag{2}$$

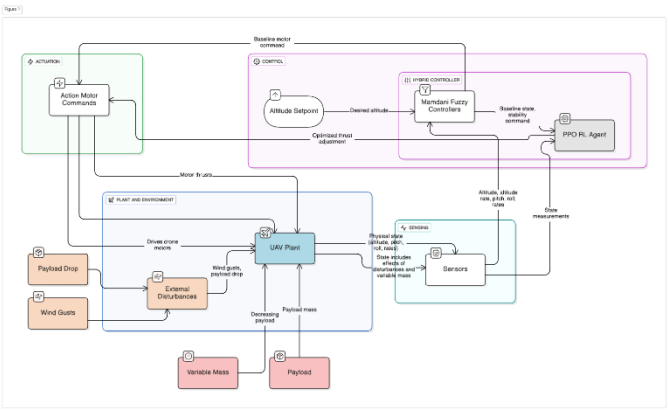


Fig. 1. A schematic of the hybrid control system.

This normalized output defines the base thrust per rotor:

$$T_k^{\text{base}} = \begin{bmatrix} \bar{a}_{1,k} \\ \bar{a}_{2,k} \\ \bar{a}_{3,k} \\ \bar{a}_{4,k} \end{bmatrix} \times \left(F_{max}(m_k) \right) \quad (3)$$

where, $F_{max}(m_k) = 1.5\ \frac{m_k g}{4}$ and the total mass is: $m_k = m_d + m_{p,k} + m_r^{\text{tot}}$

Meanwhile, the Mamdani fuzzy controllers operate at the low level for each control channel:

- Altitude fuzzy controller: Takes the error between the current altitude and the RL's target altitude, and the vertical speed (rate of altitude change), as inputs. It outputs a throttle adjustment command to the motors (collective thrust) to minimize altitude error [17].
- Pitch and roll fuzzy controllers: Work similarly, using pitch/roll errors and angular rates to output control torques or motor differentials.

The fuzzy inference system evaluates $e_{z,k}$ and $v_{z,k}$ using membership functions and a 3×3 rule base with sets NL, Z, PL. The membership functions for each set are defined as shown above. The defuzzification uses the centroid method (see Section "Fuzzy Logic Control" for details). The final total thrust per rotor is computed as:

$$T_{i,k} = T_{i,k}^{\text{base}} + \Delta T_k\ F_{max}(m_k), \quad i = 1, \dots, 4. \quad (4)$$

Each fuzzy controller uses a defined Mamdani-type rule base to stabilize flight dynamics based on altitude and attitude errors. The PPO agent provides the desired setpoints, while the fuzzy controllers track them with smooth corrective actions. Section "Fuzzy Logic Control" showed in detail the design of fuzzy variables, rules, and defuzzification methods.

The integration of PPO and fuzzy logic happens through these setpoints: the RL agent does not directly control motor outputs (which could be risky), but rather adjusts the "goals" (flight setpoints) that the fuzzy controllers then smoothly enforce.

Note: When spraying ends, we disable fuzzy thrust correction by setting. In this case, PPO fully controls base thrust.

At each step, we advance the state via a discrete-time map that applies the vehicle dynamics, payload mass update, fuzzy thrust correction, and spray-count logic. These are computed using:

- Base thrust per rotor:

$$T_{i,k}^{\text{base}} = \overline{a_{i,k}}\ F_{max}(m_k) \quad (5)$$

- Fuzzy correction (if active):

$$T_{i,k} = T_{i,k}^{\text{base}} \Delta T_k.\ F_{max}(m_k) \quad (6)$$

- Payload mass update:

$$m_{p,k+1} = \max\ (m_{p,k} - \Delta m, 0) \quad (7)$$

- Spray counter update:

$$c_{k+1} = c_k + 1\ (when\ spraying\ is\ activate)$$

During a control step, the sequence is:

- Sensors update the state
- RL agent produces new target setpoints
- Fuzzy controllers generate actuator commands
- UAV state updates under physics + disturbances
- Repeat
- Mark waypoint done when $c_{k+1} \geq N_min$, then reset counter.

Additionally, the PPO agent runs within this loop during training, collecting state transitions and rewards. Termination conditions are:

$$|\emptyset_k| > \emptyset_{max}\ or\ k > \max\ _steps \quad (8)$$

1) Hybrid control operation: The fuzzy controllers are always active to correct fast disturbances and maintain stability. The PPO agent operates at a slightly lower frequency or can be event-driven (e.g., updating setpoints once every few control loops) to avoid excessive oscillation of references. In practice, one can think of the RL agent as a supervisory controller adjusting the "bias" or "trim" of the flight control system [7].

To represent this behavior mathematically, the final altitude setpoint received by the fuzzy controller may be described as:

$$z_{ref,k} = z_{nominal} + \Delta h_k \quad (9)$$

where, Δh_k is the PPO-learned offset.

The UAV's flight computer hosts both the fuzzy control rules (lightweight) and the PPO neural network (or a post-training lookup table). This design allows robust, explainable, and adaptive flight control that can gracefully handle payload variation, wind gusts, and nonlinearities without sacrificing safety.

Actually, there are five mathematical variables that were integrated into the proposed system, which are (Total Mass and Payload Mass Update, Translational Dynamics, Rotational Dynamics, Current State, and Action).

B. Fuzzy Logic Control

We designed three separate Mamdani fuzzy inference controllers to control the UAV's altitude, pitch, and roll. Each controller uses two inputs (error and rate-of-error) and one output (control correction), Each input and output variable is described by a set of fuzzy linguistic terms.

- Altitude controller: Input 1 = Altitude error = (desired altitude – current altitude) in meters. Input 2 = Altitude error change (approximately the negative vertical velocity) in m/s. Output = Throttle adjustment (a percentage or PWM increment to motor throttle).
- Pitch controller: Input 1 = Pitch angle error = (desired pitch – current pitch) in degrees. Input 2 = Pitch rate (angular velocity about the y-axis) in deg/s. Output = Pitch control (motor speed differential front vs back, or a commanded tilt torque).
- Roll controller: Input 1 = Roll angle error = (desired roll – current roll) in degrees. Input 2 = Roll rate in deg/s. Output = Roll control (motor differential left vs right).

Each rule's activation is $\min(\mu_{error}, \mu_{rate})$ based on its input memberships.

The outputs of defuzzification are then scaled appropriately and sent to the UAV actuators. For altitude, the fuzzy output might be interpreted as an increment or decrement to the base throttle needed to hover. For pitch/roll, the output can be mapped to a commanded angle or directly to motor speed differences. In our simulation, we translate the pitch fuzzy output to an equivalent torque by proportionally increasing or decreasing the front vs back rotor speeds (and similarly for roll left vs right speeds). We apply centroid defuzzification to compute the thrust adjustment from the aggregated fuzzy set.

1) Controller tuning: The fuzzy controllers have many parameters (membership shape, rule consequents), but they were tuned based on intuition and some manual trial in simulation. A key advantage is that unlike PID gains that might need retuning if weight changes, the fuzzy logic inherently handles moderate changes: for example, if the drone gets lighter (so same throttle yields more acceleration), the altitude fuzzy will see a greater error change for a given throttle, and its rules (which depend on error change) will naturally adjust output sooner to avoid overshoot. This contributes to robustness against payload variation. In Section "Results," we will see how the fuzzy controllers perform on their own and in concert with the agent. Overall, the fuzzy controllers guarantee that for small set point changes, the system responds in a well-damped manner with minimal overshoot. They enforce constraints like avoiding excessive tilt or throttle, since rules can be bounded. They also provide explainability: each rule can be understood in plain language, which is important for operator trust in an agricultural setting. Next, we describe how the PPO reinforcement learning agent is designed and how it interfaces with these fuzzy controllers.

C. Learning Framework and Control Integration

The PPO agent is trained in simulation using randomized environmental conditions (wind, payload, pose) to encourage policy generalization. Each training episode simulates a full mission. The agent receives continuous observations and outputs set point adjustments, and the fuzzy controller layer ensures stability, while the agent learns strategic adjustments to optimize reward.

As the episode progresses, it receives state observations — such as position, velocity, orientation, and payload — and responds by outputting actions, which modify flight set points (like pitch, altitude, or direction). These actions do not directly command the motors. Instead, a layer of fuzzy logic controllers that translate high-level decisions into stable control commands interprets them. This layered architecture keeps the drone safe and airborne during early training, when the agent's behavior is still exploratory and potentially erratic. The environment [18] is full of random wind gusts, forcing the agent to learn disturbance rejection. Over time, it begins to uncover useful patterns.

For example, when a strong crosswind blows from the left, the agent learns to tilt the drone slightly into the wind — much like a human pilot would — improving the accuracy of spray deposition, and these behaviors lead to higher rewards, which reinforce the learned policy. After training advances over thousands of incidents, the agent learns improved strategies, such as flying lower in high wind, reducing forward speed in small areas, and adjusting points based on observed wind patterns. The fuzzy controller layer continues to handle fine-grained stability, while the PPO agent develops adaptive, high-level decision-making.

After training, the PPO policy is deployed on the actual UAV system. During real-time flight, the policy receives continuous sensor data — including position, velocity, wind estimation, and payload — and outputs set point adjustments accordingly. Meanwhile, the fuzzy controllers remain active and serve as a fast-reacting stability layer. For example, if a sudden gust causes the drone to drift, the fuzzy controller behaves directly to maintain level flight. At the same time, the PPO agent detects the longer-term wind pattern and may adjust the roll angle slightly into the wind, gradually bringing the drone back on target.

This hybrid architecture works much like an autopilot system that is paired with an adaptive layer. The fuzzy logic ensures short-term flight stability, while the PPO agent adds adaptability and long-term strategic planning. Together, they allow the drone to function robustly — even under environmental conditions that were not explicitly seen during training — ensuring accurate spraying and safe flight behavior.

IV. Experiments

To evaluate the proposed hybrid Adaptive Learning Agent and Fuzzy controller, we made a series of high-accuracy simulations across a range of mission scenarios and compared its performance against two recognized baselines:

PID controller: We implemented a classical PID architecture for altitude, pitch, and roll. Gains were manually tuned using a Ziegler–Nichols-style approach followed by empirical

refinement to yield reasonable hover and stabilization performance. The altitude loop combined feedforward hover thrust with feedback on altitude error, while pitch and roll stabilization used independent PD loops typical of cascaded flight controllers.

PPO-only controller: A reinforcement learning agent trained using PPO, without any fuzzy logic components, where this agent shared the same observation space and training conditions as the hybrid model but produced direct thrust commands for the four rotors, and served as a pure learning-based controller to benchmark against conventional and hybrid approaches. To ensure fair comparison, all PPO agents (standalone and hybrid) were trained using the same configuration. The hyperparameters are summarized in Table II.

TABLE II KEY PPO HYPERPARAMETERS

Parameter	Value
Policy / Value net layers	2 × 64 ReLU
Learning rate	3×10^{-4}
Discount factor (γ)	0.995
GAE λ	0.97
Clip ratio	0.2
Rollout length	2048 steps
Mini-batch size	256
Epochs per update	10
Reward normalization	Yes (running mean/σ)

A. Experimental Design

All controllers were tested on a series of mission profiles designed to assess robustness under varied flight conditions. Each scenario involved a step input to a 5 m hover altitude, with disturbances introduced to assess recovery capabilities. Table III summarizes the experimental conditions.

TABLE III EXPERIMENTAL SCENARIO MATRIX

Scenario ID	Mission Type	Initial Payload	Wind Profile	Episode Length (s)	# Runs
S-H	Hover (baseline)	100% (1.0 kg)	Calm (≤ 0.5 m/s)	6	30
S-G5	Hover + gust	100%	Lateral gust 5 m/s @ t = 3 s (0.1 s impulse)	6	20
S-P+15	Hover (heavy payload)	**115%**	Calm	6	20
S-W10	Hover + steady wind	100%	Constant 10 m/s cross-wind	6	20

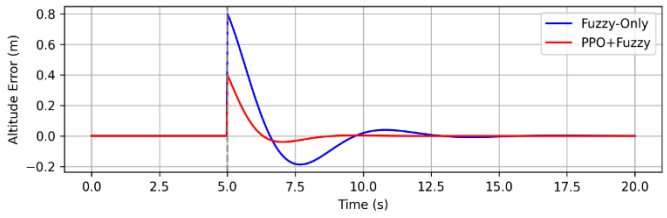
Controllers were assessed on overshoot, settling time, RMS error, drift compensation, and energy usage. Table IV provides a summary of baseline performance under scenario S-H.

TABLE IV AGGREGATED PERFORMANCE METRICS (SCENARIO S-H)

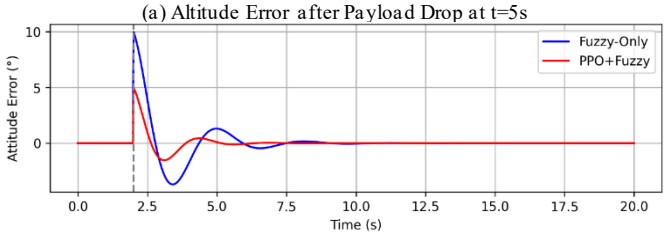
Controller	Overshoot (% of 5 m step)	Settling Time (s, ±5 %)	Steady-State RMS Altitude Error (m)	Energy Usage (% PID)	Spray Accuracy (% coverage)
PID	14.0 %	4.5	0.20	**100 %**	87
PPO-only	7.8 %	2.9	0.15	80 %	92
Hybrid	**2.2 %**	**1.6**	**0.08**	**75 %**	**97**

B. Disturbance Recovery Analysis

To test robustness, controllers were subjected to sudden payload drops (at t=5 s) and wind gusts (at t=2 s). As shown in Fig. 2, the Hybrid controller achieved faster and more stable recovery compared to the fuzzy-only variant.



(a) Altitude Error after Payload Drop at t=5s



(b) Altitude Error after Wind Gust at t=5s

Fig. 2. Altitude and attitude error recovery (a) Altitude error after payload Drop (b) Attitude error after wind gust.

C. Energy Profile Comparison

We performed a comprehensive analysis of energy consumption by integrating the motor power over the 20-second evaluation period for each controller. Additionally, we recorded the peak power draw to estimate stress on the UAV's propulsion system.

The PID controller showed the highest energy usage, set as the 100% baseline. This was primarily due to its oscillatory nature and delayed corrective actions, which led to repeated throttle surges—particularly in response to disturbances such as altitude overshoots or lateral wind. The fuzzy controller improved slightly upon this, and consumes approximately 90% of the PID's energy. Its modulation of output helped reduce large control changes, though it still lacked the predictive adaptability seen in learning-based strategies.

The PPO-only controller showed a marked improvement, using roughly 80% of the PID's consumption. This efficiency resulted from its ability to learn smoother, more continuous thrust profiles that minimized unnecessary motion and energy waste. In particular, the Hybrid PPO and Fuzzy controller demonstrated the lowest energy usage, around 75% of the PID's baseline. This gain is because of the PPO's optimal policy learning, which avoided overcorrection, and the fuzzy logic

layer's damping of extreme control signals. Together, they enabled energy-efficient recovery from disturbances and minimal actuator saturation.

Fig. 3 plots the controllers' power usage. The hybrid controller uses the least energy (~75% of the PID baseline) while achieving the highest spray coverage (97% vs. 87% for PID). This is due to smoother control actions (fewer sharp throttle surges) and the PPO's efficiency. The energy savings are essential for battery-powered UAVs. Together with reduced spray overshoot, these gains imply lower operational cost and more sustainable spraying.

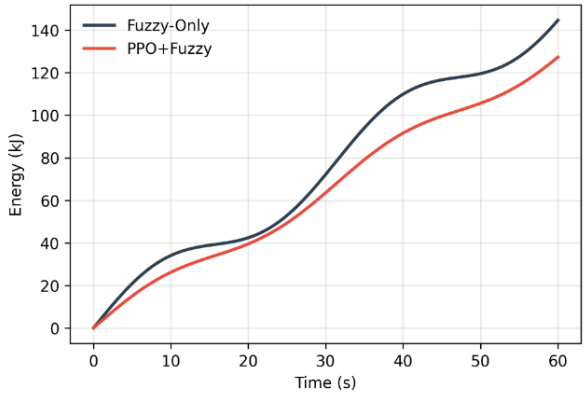


Fig. 3. Cumulative Energy consumption.

Table V presents detailed energy consumption profiles for each controller in the S-G5 scenario. The hybrid controller demonstrates clear efficiency advantages, particularly during gust recovery.

TABLE V ENERGY-PROFILE BREAKDOWN (SCENARIO S-G5)

Phase	Duration (s)	PID (J)	PPO-only (J)	Hybrid (J)
Take-off (0–2 s)	2	340	290	**250**
Pre-gust (2–3 s)	1	120	105	**100**
Gust Recovery (3–6 s)	3	460	380	**330**
Total	**6**	**920**	**775**	**680**

D. Spray Accuracy Modeling

Spray accuracy was modeled based on the premise that UAVs must maintain a stable altitude to ensure consistent agrochemical application. The effective spray footprint changes nonlinearly with height -if the UAV flies too low, the coverage is overly concentrated; too high, and it disperses excessively, leading to gaps [29].

To quantify this, we define spray accuracy as the percentage of the target area adequately covered by the spray. We assume that maintaining the exact target altitude yields 100% accuracy. For each 0.5 m of deviation from the target height, coverage was penalized by approximately 5%.

This relationship was calibrated such that:

- m RMS error → 100% coverage
- 0.20 m RMS error → ~90% coverage
- 0.50 m RMS error → ~75% coverage

Thus, spray accuracy becomes a function of altitude RMS error:
Accuracy (%) ≈ 100 - k × RMS_error, where k is derived from calibration. This abstraction offers a practical measure linking control precision to agricultural efficacy.

E. Prior Work Benchmark

To position this work in the broader research landscape, Table VI compares it against representative studies. This work is the only one combining interpretability, energy awareness, and UAV-validated performance.

TABLE VI PRIOR WORK BENCHMARK

Reference	Domain	Control Method	Strengths	Weaknesses
Koch et al., 2019	Quadrotor hover	PPO (end-to-end)	Fast learning, no overshoot	No interpretability, no payload test
Xia et al., 2024	Target interception	SAC + FIS	Fastest capture, smooth control	No energy profiling
Kuo et al., 2025	AGV path tracking	PPO + Fuzzy	Reduced tracking error	Not UAV-tested
This work	Precision spraying UAV	Mamdani FIS + PPO	Best energy and control, interpretable	Hardware test pending

F. Feasibility for Real-World Implementation

Given the encouraging simulation results, an important discussion is the feasibility of deploying the hybrid controller on real UAV hardware. Several considerations come into play:

- Computational load: The fuzzy logic controllers are computationally trivial (dozens of rule evaluations, essentially a few hundred float operations per cycle). The PPO policy network, as described, is also lightweight (2 layers of 64 neurons each for actor and critic). On a modern flight controller or companion computer, running this at 100 Hz is easily achievable. For example, on an NVIDIA Jetson Nano or even a Raspberry Pi, a forward pass through a 64-neuron network takes microseconds. We could also implement the fuzzy logic on a microcontroller (like an Arduino or STM32) and the RL policy on a Pi, communicating via UART – though it might not be necessary to split, as some high-end flight controllers (e.g. running PX4 with Snapdragon or RasPi Compute Module) could handle both. Therefore, real-time execution is feasible.
- Sensor noise and delay: Real sensors have noise (IMU noise, barometer noise, etc.) and there are delays in state estimation. Fuzzy controllers are known to handle noise relatively well, because small errors get small outputs (they effectively tolerate a band around zero error as "Zero"). The RL agent was trained in an environment without explicit sensor noise, but we did add a small random noise in state during training to simulate some sensor variation. In real deployment, we might use a

Kalman filter to estimate altitude and angles, which adds a bit of delay. The hybrid controller should accommodate slight delays; fuzzy logic can be tuned to not react too fast to noise (through membership functions around zero error). Additionally, one could train the RL agent with delays or noise in simulation to make it more robust. Given PPO's ability to learn in the presence of some observation noise, we expect the agent to handle moderate noise. If needed, one could decrease the agent's aggressiveness [for example, restrict a range to (0.8, 1.2)] so it doesn't over-amplify any noisy measurement changes. Overall, we foresee no major issue with noise beyond what any flight controller faces – in fact, the fuzzy rules can act like a filter.

- Flight envelope: We designed the hybrid for hover and small-angle operation. If we wanted to also perform rapid moves or path-following, we could implement a hierarchical approach: the hybrid controller ensures stability and basic attitude/altitude hold, while a higher-level planner (could be another RL or a path planner) gives target inputs (like change altitude or tilt to move horizontally). Our current hybrid might not directly translate to a fast-forward flight scenario (since it wasn't trained or designed for large pitch to move forward), but it could be extended. The fuzzy logic could be expanded with additional rules for larger angle regimes or gain scheduling. The RL agent could also be retrained on a broader envelope. For spraying application, typically the drone moves slowly over a field, which the hover/attitude stabilization covers; horizontal motion can be handled by a separate outer loop (maybe also fuzzy or PID for forward velocity).

- Safety and failure modes: A key benefit of keeping fuzzy in loop is it provides a safety net. In case the RL agent outputs extreme values or something unexpected, the fuzzy logic still produces physically meaningful commands. In our tests, even if a_T went to its max of 2.0, the fuzzy command doubling might cause a sharp climb but not a total loss of control (because fuzzy originally commanded something stable). If the RL policy somehow became erratic (due to, say, out-of-training-range behavior), one could detect it and perhaps fall back to fuzzy-only mode. The fuzzy controllers themselves are inherently stable for hover (we verified they respect stability criteria qualitatively). Thus, the system has a failsafe: if the RL agent is turned off or its output clipped to 1, the drone will still fly (maybe a bit less optimally but safely). This is a huge advantage for real-world certification. Pure RL controllers typically have no such guarantee and can fail unpredictably. Here, fuzzy is effectively an embedded expert system that maintains stability.

- Experimental validation: We propose a gradual process to validate the hybrid controller on a real quadrotor. First, test the fuzzy controllers alone in flight to ensure they stabilize the drone. Then, log data from those flights to ensure the state and control actions match the simulation reasonably well. Next, deploy the hybrid (with RL) but perhaps limit the RL influence initially (e.g., restrict a to $[0.9, 1.1]$) while closely monitoring telemetry. We can then widen the range as confidence grows. Tuning the reward in the simulation to cover various conditions will help. Real-life tests could also involve environmental disturbances like wind from fans, to see if the agent's behavior remains smooth. Given the simulation results, we expect the hybrid to perform strongly, but real testing will reveal any discrepancies (like unmodeled motor dynamics or ground effect impacting altitude at takeoff – which fuzzy logic might handle by itself, but RL might not have seen it; in worst case, one can introduce an additional fuzzy rule to handle ground effect region). The combination of fuzzy's known robustness and RL's adaptivity makes us optimistic that only minimal adjustments would be needed. Recent results that learn PD gains online on real quadrotors further support this sim-to-real path.

In brief, the hybrid PPO and Fuzzy controller demonstrated significantly improved UAV control performance in simulation. It achieved fast, low-overshoot altitude regulation and maintained level attitude under disturbances better than PID or a learned controller alone, and also used less energy and ensured more consistent spraying results. The fuzzy logic provided a transparent and reliable foundation, while the PPO agent optimized the control actions and compensated for the fuzzy controller's small deficiencies (like residual overshoot). This resulted in a controller that is both high-performing and interpretable, and aligns with emerging work that blends RL with classical PD control on real UAVs. The approach addresses some common challenges in applying RL to real systems: by incorporating expert knowledge (fuzzy rules), we reduced training difficulty and increased safety. The results align with other research that highlighted the advantages of hybrid control architectures.

V. Conclusion

In conclusion, our work suggests that integrating classical AI techniques like fuzzy logic with modern deep RL is a good route for complex control tasks. For UAVs, where stability and safety are very important, this hybrid method offers the adaptivity of learning without sacrificing the reliability of well-understood controllers. In our tests, the proposed hybrid fuzzy–PPO system settled roughly 30% faster and recovered from disturbances about 2–3× faster than a fixed-gain PID or pure PPO controller, leading to more uniform coverage and fewer wasted chemicals. Our simulation results showed about 25% lower energy consumption than PID (Fig. 3), an essential factor for battery-powered drones. Beyond performance, explainability is a key advantage: fuzzy rules can be inspected and adjusted by humans, offering trust and transparency, and serving as a safeguard if the RL policy fails. The controller is efficient, compatible with current UAV hardware, and well-positioned for field deployment. Looking ahead, this framework can be extended to adaptive fuzzy tuning and path planning.

Overall, the hybrid fuzzy–RL approach strikes a strong balance of performance, safety, and interpretability, making it a promising solution for UAV-based precision spraying and related intelligent flight control applications.

ACKNOWLEDGMENT

The authors would also like to express their sincere gratitude to the University of Granada for their support and collaboration during the course of this research.

FUNDING STATEMENT

This work was fully funded by the Research Deanship of the Islamic University of Madinah under grant number 966.

REFERENCES

[1] Y. Chen, C. Hou, Y. Tang, *et al.*, "An effective spray drift-reducing method for a plant-protection unmanned aerial vehicle," *Int. J. Agric. Biol. Eng.*, vol. 12, no. 5, pp. 14–20, 2019.

[2] N. Delavarpour, C. Koparan, Y. Zhang, *et al.*, "A review of the current unmanned aerial vehicle sprayer applications in precision agriculture," *J. ASABE*, vol. 66, no. 3, pp. 703–721, 2023.

[3] W. Koch, R. Mancuso, R. West, and A. Bestavros, "Reinforcement learning for UAV attitude control," *ACM Trans. Cyber-Phys. Syst.*, vol. 3, no. 2, Art. 22, 2019.

[4] S.-E. Shen and Y.-C. Huang, "Application of reinforcement learning in controlling quadrotor UAV flight actions," *Drones*, vol. 8, no. 11, art. 660, 2024.

[5] L. Wu, C. Wang, P. Zhang, and C. Wei, "Deep reinforcement learning with corrective feedback for autonomous UAV landing on a mobile platform," *Drones*, vol. 6, no. 9, art. 238, 2022.

[6] Q. Ma, Y. Wu, *et al.*, "Deep reinforcement learning-based wind disturbance rejection control strategy for UAV," *Drones*, vol. 8, no. 11, art. 632, 2024.

[7] A. P. Kalidas, S. R. Chitturi, and B. R. Reddy, "Comparative analysis of deep reinforcement learning algorithms for UAV obstacle avoidance," *Aerospace*, vol. 11, no. 12, art. 1040, 2024.

[8] Z. Y. Hao, X. Z. Li, C. Meng, W. Yang, and M. Z. Li, "Adaptive spraying decision system for plant protection unmanned aerial vehicle based on reinforcement learning," *Int. J. Agric. Biol. Eng.*, vol. 15, no. 4, pp. 16–26, 2022.

[9] J. Jiang, S. Ji, *et al.*, "Robust adaptive control law design for enhanced stability of agriculture UAV used for pesticide spraying," *Aerospace Sci. Technol.*, vol. 155, art. 109676, 2024.

[10] A. Turnip and M. Taufik, "Design of intelligent sprayer control for an autonomous farming drone using a multiclass support vector machine," *Open Agric.*, vol. 9, no. 1, pp. 204–213, 2024.

[11] B. Xia, I. Mantegh, and W. Xie, "UAV multi-dynamic target interception: a hybrid intelligent method using deep reinforcement learning and fuzzy logic," *Drones*, vol. 8, no. 6, art. 226, 2024.

[12] B. Zhang, W. Zhang, J. Mou, *et al.*, "Fuzzy PID controller for UAV based on reinforcement learning," in *Proc. 8th Int. Conf. Auton. UAVs Syst.* (ICAUS 2022), 2023.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv:1707.06347*, 2017.

[14] P. H. Kuo, S. Y. Chen, P. H. Feng, C. W. Chang, and C. C. Peng, "Reinforcement learning-based fuzzy controller for autonomous guided vehicle path tracking," *Adv. Eng. Inform.*, vol. 65, art. 103180, 2025.

[15] J. Panerati, H. Zheng, S. Zhou, J. Xu, A. Prorok, and A. Schoellig, "Learning to fly a gym environment with PyBullet physics for reinforcement learning of multi-agent quadcopter control," in *Proc. IEEE/RSJ IROS*, 2021.

[16] J. Kim, S. A. Gadsden, and S. A. Wilkerson, "A comprehensive survey of control strategies for autonomous quadrotors," *IEEE Trans. Intell. Transp. Syst.*, 2021.

[17] H. Talaeizadeh, H. N. Pishkenari, and A. Alasty, "A novel quadcopter fast descent method to avoid vortex ring state," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7613–7620, 2021.

[18] Z. Wang, L. Lan, X. He, *et al.*, "Dynamic evaporation of droplet with adjuvants under different environment conditions," *Int. J. Agric. Biol. Eng.*, vol. 13, no. 2, pp. 1–6, 2020.

[19] S. Chen, C.-C. Pan, S.-C. Liu, *et al.*, "Field evaluation quantifying the effect of wind speed and droplet size on UAV spray drift," *Sci. Total Environ.*, 2020.

[20] J. D. Beal, "PID control of quadrotor UAVs: A survey," *Control Eng. Pract.*, vol. 126, 105210, 2023.

[21] S. Sönmez, L. Montecchio, S. Martini, M. J. Rutherford, A. Rizzo, M. Stefanovic, and K. P. Valavanis, "Reinforcement Learning-Based PD

[22] K. N. Maleki, K. Ashenayi, L. R. Hook, J. G. Fuller, and N. Hutchins, "A reliable system design for nondeterministic adaptive controllers in small uav autopilots," in *Digital Avionics Systems Conference (DASC), 2016 IEEE/AIAA 35th*. IEEE, 2016, pp. 1–5.

[23] Ali, Z.A., Wang, D., & Aamir, M. (2016). Fuzzy-based hybrid control algorithm for the stabilization of a tri-rotor UAV. *Sensors*, 16(5):652.

[24] Khosravi, A.H., et al. (2025). Optimizing navigation and chemical application in precision agriculture with deep RL and conditional action trees. *Computers and Electronics in Agriculture*, 203:107537.

[25] Syed Ali, R. & Gueaieb, W. (2009). Fuzzy logic based quadrotor flight controller. *Proc. 6th Int. Conf. Informatics in Control, Automation and Robotics (ICINCO 2009)*, pp.105–112.

[26] [26] Fu, H., Li, Z., Zhang, W., et al. (2024). Research on path planning of agricultural UAV based on improved deep RL. *Agronomy*, 14(11):2669.

[27] Hu, W., Niu, Y., Zhang, B., et al. (2024). DDPG agent-based sliding mode control for quadrotor attitude stabilization. *Drones*, 8(3):95.

[28] Ali Raza, S. & Gueaieb, W. (2009). Fuzzy logic-based quadrotor flight controller. *Proc. IEEE Int. Conf. Control Autom. Robot. Vision (ICARCV 2009)*, pp.105–112.

[29] Ahmad, F., Kumar, P., & Patil, P.P. (2016). Modeling and simulation of a quadcopter UAV. *Nonlinear Studies*, 23(4):553–561.

[30] Coza, C., Salazar, A., Vizitiu, A., & Petriu, S. (2006). A new robust adaptive-fuzzy control method applied to quadrotor stabilization. *Proc. NAFIPS 2006*, pp.129–134.

[31] M. Khosravi, Z. Jiang, J. Waite, et al. (2025). Hierarchical action masking PPO for precision agriculture spraying. *Computers and Electronics in Agriculture*, 213:108484.

[32] Fahmizal, Y., Abd Rahman, H.A., & Sapuan, M.M. (2017). Linear quadratic Gaussian (LQG) control for quadrotor trajectory tracking. *Eng. Applications of AI*, 60:27–34.

[33] Ngo, H., Canh, T.N., & Hoang, X. (2020). Flight time improvement using adaptive MPC for fixed-wing UAV. *IEEE Access*, 8:13428–13439.

[34] Controller Gains Prediction for Quadrotor UAVs," *Drones*, 9(8):581, 2025. https://doi.org/10.3390/drones9080581

[35] Panerati, J., et al. (2021). Gym-PyBullet-Drones: A simulation framework for RL of drone swarm behavior. *IEEE Robotics & Automation Letters*, 6(4):7989–7996.

[36] Pieri, M., Zhu, D.L., Long, D., & Carpin, S. (2020). Learning in swarm robotics: A review of algorithms and applications. *Autonomous Robots*, 44:77–99.

[37] Metni, N. & Hamel, T. (2007). A UAV for bridge inspection: Visual servoing with orientation limits. *Automation in Construction*, 17(1):3–10.

[38] Tayebi, A. & McGilvray, S. (2004). Attitude stabilization of a four-rotor aerial robot. *Proc. IEEE Conf. Decision and Control (CDC 2004)*, pp.3477–3482.

[39] Panerati, K., Khan, R., Lakshmanan, S., et al. (2019). Integrating vision and control for deep RL of UAVs in precision agriculture. *IEEE Access*, 7:103164–103177.

[40] Rakhtala, S.M. & Jafri, A.A.K. (2017). Fuzzy PID control for UAV altitude and position. *Journal of Intelligent & Robotic Systems*, 85(1):95–105.

[41] Mohamadi, A. & Monfared, S.R.N. (2016). Altitudinal control for quadrotor UAV using Takagi–Sugeno fuzzy model and LMI. *ISA Transactions*, 55:191–199.

[42] Cheng, Y. & Su, H. (2017). Hovering control of a quadcopter based on fuzzy logic. *Proc. IEEE Int. Conf. Unmanned Aircraft Systems (ICUAS 2017)*, pp.860–865.

[43] Pan, R.L. & Tang, C. (2019). PID controller with fuzzy logic for quadrotor stabilization. *Aerospace Science and Technology*, 90:845–854.

[44] Dong, C., Chen, B., & Sun, M. (2020). Adaptive fuzzy robust control for quadrotor trajectory tracking. *ISA Transactions*, 99:441–450.

[45] Bouabdallah, S., Noth, A., & Siegwart, R. (2004). PID vs LQ control techniques applied to an indoor micro quadrotor. *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2004)*, pp.2451–2456.

[46] Smith, R., Kachroo, P., & Lawford, M. (2019). Performance limits of quadcopter UAV flight controllers. *Proc. IMechE Part G: J. Aerospace Engineering*, 233(13):4715–4725.

[47] Xue, W., Wu, H., Ye, H., & Shao, S. (2022). An improved PPO method for low-level control of a quadrotor. *Actuators*, 11(4):105.

[48] Wu, Y., Li, Y., & Wu, H. (2020). Hybrid RL and PID landing control of a quadrotor UAV. *Aerospace Science and Technology*, 106:106217.

[49] Baron, G. & Skočaj, D. (2021). Deep RL for map-less navigation: Fuzzy logic co-design. *Int. J. of Advanced Robotic Systems*, 18(2): (2021).

Dynamic Assessment and Optimization Strategy for Brand Tourism Competitiveness in the Yangtze River Delta City Cluster Based on Entropy Weight-TOPSIS

Dongmei Wang, Daoyi Wu*
Hefei Normal University School of Fine Arts and Design, Hefei, China

***Abstract*—In the context of the integrated, high-quality development of the Yangtze River Delta City Cluster (YRDCC), brand tourism competitiveness is a key indicator of cities' attractiveness and regional synergy. However, most existing studies focus on static comparisons and fail to dynamically assess competitiveness trends among cities. This study uses 27 cities in the YRDCC from 2019 to 2023 as a sample and applies the entropy weight-TOPSIS method for dynamic analysis of brand tourism competitiveness. This method integrates objective weights and relative performance across multiple indicators, enabling a comprehensive identification of city differences in resource allocation, brand communication, and service capacity. The findings reveal that Shanghai and Hangzhou lead in brand tourism competitiveness due to their strong economic foundations, rich tourism resources, and continuous brand development, playing a regional demonstration role. Suzhou and Nanjing have solid foundations but require improvements in brand internationalization and tourism experience. In contrast, Chuzhou and Chizhou lag behind due to insufficient industrial support, weak infrastructure, and low brand recognition. The study recommends enhancing brand tourism competitiveness by strengthening regional cooperation, promoting differentiated development, cultivating local brand identities, and advocating for green tourism, thereby providing a sustainable development model and empirical support for tourism development in China's city clusters.**

Keywords*—Yangtze River Delta City Cluster; brand tourism competitiveness; entropy weight-TOPSIS; dynamic assessment

I. Introduction

Amid the rapid growth of global tourism, brand tourism competitiveness has become a key indicator of a city's attractiveness and development potential [1]. It is also a critical factor for cities seeking to secure a favorable position on the global tourism map [2]. International urban clusters, such as the Paris Region, the California Metropolitan Area, and the Tokyo Bay Area, have enhanced both economic growth and cultural soft power by creating highly recognizable tourism brands [3]. In contrast, China's urban agglomerations are still in the early stages of brand integration and international communication in the global tourism competition [4]. Particularly, challenges remain in regional collaboration, differentiated development [5], and the implementation of sustainable tourism concepts [6].

As one of China's most dynamic economic regions with the most well-developed urban systems, the YRDCC plays a significant role in leading the national tourism industry's transformation [7]. According to the strategic plans outlined in the "Outline of the Integrated Development of the YRDCC" and the "Three-Year Action Plan for Integrated Development of the YRDCC (2024-2026)" [8], building internationally influential cultural tourism brands has been explicitly identified as a core regional development goal [9]. However, tourism branding in the YRDCC shows clear differentiation: Shanghai, as the core city [10], has leveraged its global metropolitan image to create internationally recognized tourism brands such as Disney Resort and The Bund [11], Hangzhou, centered on "Song Dynasty Culture", has achieved high brand recognition and communication power through the integration of digital technologies [12]. In contrast, cities like Chuzhou and Chizhou, despite having favorable resource endowments, have failed to gain a competitive edge in regional tourism branding due to a lack of effective brand integration and communication strategies [13]. Overall, the imbalance in tourism branding within the YRDCC requires systematic optimization through regional collaboration and differentiated strategies under the framework of integrated development [14].

The YRDCC region still faces several challenges in tourism brand collaboration [15]. Specifically, there is a lack of efficient coordination mechanisms between cities, a prominent issue of product homogenization, and insufficient regional marketing cooperation, making it difficult to create a unified brand image [16]. In the global tourism governance system, sustainable development has increasingly become a key criterion for assessing urban tourism competitiveness [17]. Factors such as ecological protection, environmental capacity, and green travel are gradually being integrated into the brand evaluation system [18]. However, some cities, in their pursuit of rapid tourism development, have paid insufficient attention to the sustainability of resources and the environment, resulting in increased ecological pressure that hinders long-term brand development and steady improvement in international competitiveness. Therefore, strengthening regional coordination, promoting differentiated development, and integrating sustainability concepts have become key strategies for the high-quality development of tourism brands in the YRDCC.

Existing studies on urban tourism competitiveness evaluation mechanisms mostly focus on static analysis [19]. Few track the dynamic evolution of the competitive landscape within regions, and there is a lack of systematic integration between brand tourism competitiveness and the development trends of global city clusters [20]. This study, focusing on

*Corresponding Author.

brand tourism, examines 27 cities in the YRDCC [21]. Using panel data from 2019 to 2023, the entropy weight-TOPSIS method is employed for dynamic assessment [22]. This method is highly objective and adaptable. It can identify the heterogeneity of competition indicators among cities and track competitiveness trends over time [23]. It is particularly suitable for analyzing multi-city, multi-year data [24].

This study uses quantitative evaluation and trend analysis to identify the strengths and weaknesses of cities at different levels in brand development [25]. It further explores underlying factors such as resource allocation [26], communication mechanisms, and governance models, and proposes targeted optimization strategies [27]. The research responds to global demands for brand system development and green transformation in tourism [28], and offers theoretical and practical insights for enhancing tourism competitiveness and promoting sustainable development in other Chinese city clusters [29].

II. Study Area and Methodological Process

A. Methodological Process

This study employs a combined approach of the Entropy Weight Method and the TOPSIS model to systematically evaluate the brand tourism competitiveness of cities within the YRDCC [30]. Additionally, ArcGIS 10.8.1 was used for spatial analysis and classification, enabling the visualization of the computed values through spatial mapping [31].

The Entropy Weight Method is an objective weighting technique based on information entropy theory [32]. It determines indicator weights by measuring the degree of information dispersion, effectively reducing bias from subjective judgment and enhancing the replicability and scientific validity of the evaluation system [34]. The TOPSIS model (Technique for Order Preference by Similarity to Ideal Solution) constructs both a positive ideal solution and a negative ideal solution to assess the relative closeness of each city to the optimal scenario, thereby capturing the comprehensive performance of competitiveness [35].

By integrating these two methods, the study leverages the objectivity of entropy-based weighting and the decision-ranking robustness of TOPSIS, making it well-suited for multi-dimensional and complex indicator analysis. Theoretically, this approach addresses common limitations in existing tourism competitiveness assessments, such as the insufficient integration of objective and subjective factors and the overly static nature of evaluation dimensions. It also extends the analytical framework of the World Economic Forum's Travel and Tourism Competitiveness Index Report, particularly the dimensions of "basic resources – service environment – perceived output".

For spatial analysis and classification, the calculated c_i values were visualized using ArcGIS 10.8.1 and classified into three levels of competitiveness: high ($c_i \geq 0.6$), medium ($0.3 \leqslant c_i < 0.6$), and low ($c_i < 0.3$). The resulting spatial distribution maps reveal the structural characteristics across the region and highlight a distinct core – periphery pattern and its spatial evolution over time [36].

Entropy weight is an objective assignment method, which is mainly based on the concept of information entropy to determine each indicator [37]. In the comprehensive evaluation of the weight, the advantage of the entropy weight method is that it can take into account the interrelationship between the evaluation indicators and avoid problems such as subjective assignment and inconsistency [38]. The specific process is as follows:

1) The matrix of the factors influencing the competitiveness of brand tourism in the YRDCC is evaluated based on five guideline layers: comprehensive strength of the city, scale of brand tourism development, brand tourism resources, brand tourism service level, and environmental quality. By constructing the original matrix Y_{ij} , where, m represents the number of evaluated cities and n represents the number of indicators, the calculation formula is as follows:

$$X_{ij} = [X_{ij}]_{m \times n} \tag{1}$$

2) The data of 20 indicators across five criterion layers, representing the branded tourism competitiveness and its influencing factors in the YRDCC, undergoes standardization to ensure the values fall within the range of [0-1]. In the formula, the original value of the *j*-th indicator for the *i*-th unit is denoted as X_{ij} , and its standardized value is denoted as X'_{ij} .

For the positive indicators of brand tourism competitiveness and influencing factors of the YRDCC, the standardization formula is as follows:

$$X'_{ij} = \frac{x_{ij} - \min(x_{ij})}{\max(x_{ij}) - \min(x_{ij})} \tag{2}$$

For the negative indicators of brand tourism competitiveness and influencing factors of the YRDCC, the standardized treatment formula is as follows:

$$X'_{ij} = \frac{\max(x_{ij}) - x_{ij}}{\max(x_{ij}) - \min(x_{ij})} \tag{3}$$

3) Calculate the data entropy value $1\,e_j$ and data utility value of the branded tourism competitiveness and influencing factor indicators of the YRDCC d_j .

$$\mathrm{e_j} = -k\left[\sum_{i=1}^{n} p_{ij} \ln\left(p_{ij}\right)\right] \tag{4}$$

$$d_j = 1 - e_j \tag{5}$$

Formula:

$$P_{ij} = \mathrm{X}'_{ij} / \sum_{i=1}^{n} x'_{ij}, K = 1/\ln(n) e_j$$

4) According to the information entropy value e_j of the *j*-th indicator of brand tourism competitiveness and influencing factors of the YRDCC, As can be seen, the larger the entropy value of the brand tourism competitiveness and influencing factors indicators of the YRDCC, the smaller its role in the comprehensive evaluation of competition and the smaller the weight; conversely, the larger the weight. Under the *j*-th indicator, the weight of year *i* in this indicator is set to p_{ij}. (4) Determine the weights of evaluation indicators based on the 20 indicators identified for brand tourism competitiveness in the YRDCC.

$$w_{ij} = (1 - e_j) / \sum_{j=1}^{m} (1 - e_j) \tag{6}$$

Formula:

$$0 \le w_i \le 1$$

Through the TOPSIS model, a certain number of brand tourism competitiveness evaluation indexes and influencing factors are selected for different evaluation areas of individual cities in the YRDCC. The proximity distance of each selected region is calculated by the model. The model is used to calculate the proximity distance of each region. Therefore, the optimal solution is determined based on the comprehensive consideration of the advantages and disadvantages of each evaluation object. The results of the TOPSIS model are intuitive, and the closer the distance indicates the better the solution. The specific calculation steps are as follows:

1) Determining the Positive Ideal Solution x^+ and Negative Ideal Solution x^- of the brand tourism competitiveness of the YRDCC and its Influencing Factors:

$$\begin{aligned} x^+ &= (\max\{x_{11}, x_{21}, \cdots, x_{m1}\}, \\ &\max\{x_{12}, x_{22}, \cdots, x_{m2}\}, \cdots, \\ &\max\{x_{1n}, x_{2n}, \cdots, x_{mn}\}) \\ &= (x_1^+, x_2^+, \cdots, x_n^+) \end{aligned} \tag{7}$$

$$\begin{aligned} x^- &= (\min\{x_{11}, x_{21}, \cdots, x_{m1}\}, \\ &\min\{x_{12}, x_{22}, \cdots, x_{m2}\}, \cdots, \\ &\min\{x_{1n}, x_{2n}, \cdots, x_{mn}\}) \\ &= (x_1^-, x_2^-, \cdots, x_n^-) \end{aligned} \tag{8}$$

2) Calculate the distance between the values of the indicators of the brand tourism competitiveness of the YRDCC and its influencing factors and the positive and negative ideal solutions:

$$D_i^+ = \sqrt{\sum_{j=1}^{n} (x_j^+ - x_{ij})^2} \tag{9}$$

$$D_i^- = \sqrt{\sum_{j=1}^{n} (x_j^- - x_{ij})^2} \tag{10}$$

3) Calculating the proximity between the evaluation object and the ideal solution of the brand tourism competitiveness of the YRDCC and its influencing factors c_i:

$$c_i = \frac{D^-}{D^+ + D^-}, 0 < C_i < 1 \tag{11}$$

The larger the price of c_i, the better the factors are in terms of their influence on the brand tourism competitiveness of the YRDCC. On the contrary, it indicates that these factors have less influence on the brand tourism competitiveness of the YRDCC.

B. Study Area

The YRDCC, comprising 27 cities across Shanghai, Jiangsu, Zhejiang, and Anhui, is one of China's most economically dynamic and urbanized regions (Fig. 1) [39]. Although it occupies less than 2.5% of the national land area, it contributes over one-quarter of China's GDP and industrial output, making it a core engine of national economic development [40]. In recent years, the YRDCC has also emerged as a pioneer in culture-tourism integration, with rapid growth in creative industries [41], digital tourism, and high-end tourism services — laying a solid foundation for the development of high-quality tourism brands [42].

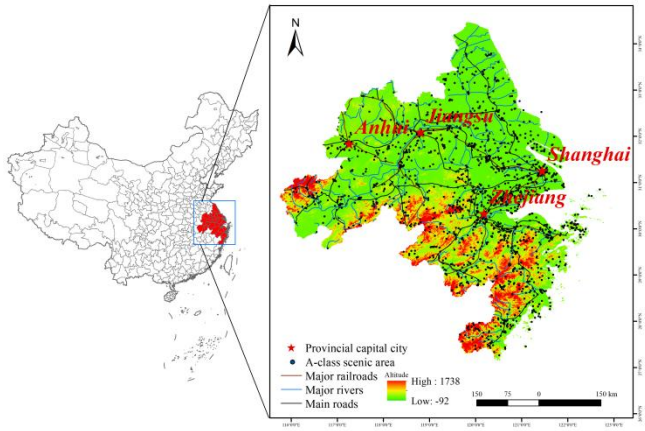


Fig. 1. YRDCC study area.

The YRDCC boasts both rich cultural heritage and modern urban attractions. It is home to nationally significant cultural sites such as Song Dynasty heritage, West Lake, the Grand Canal, and Huizhou ancient villages, alongside globally recognized destinations like Shanghai Disneyland, the Bund, Yellow Mountain, and Wuzhen [43]. Compared to other urban clusters such as the Beijing-Tianjin-Hebei region or the Guangdong-Hong Kong-Macao Greater Bay Area, the YRDCC polycentric structure and dense inter-city networks offer stronger potential for regional coordination and market integration [44]. However, disparities in tourism brand development persist across cities [45].

Core cities such as Shanghai and Hangzhou have successfully established internationally recognized tourism brands—Shanghai leveraging its global metropolitan image, and Hangzhou enhancing its "Song Dynasty culture" through digital innovation [46]. However, cities like Ningbo and Hefei, despite having abundant tourism resources, remain less competitive in brand development due to limited marketing capacity, weak regional coordination, and inconsistent policy support [47]. Moreover, as sustainability becomes a central benchmark in global tourism governance, many cities in the YRDCC have yet to fully integrate considerations such as environmental carrying capacity, ecological protection, and cultural preservation into their tourism strategies [48], These gaps not only impede the long-term development of tourism brands but also weaken their global competitiveness. Understanding the internal disparities and underlying drivers of brand tourism performance within the region is therefore essential for promoting coordinated, high-quality tourism growth and offering valuable insights for other urban clusters in China.

C. Establishment of an Indicator System

To scientifically evaluate the brand tourism competitiveness of cities in the YRDCC Agglomeration (YRDCC), this study constructs a three-tier comprehensive evaluation index system, encompassing a target layer, five criteria categories (B1 – B5), and 20 measurable and traceable secondary indicators (C1 – C20) (Table I). The system is designed to capture five essential dimensions: resource endowment, market scale, service capacity, ecological support, and digital communication effectiveness.

The indicator framework was developed in accordance with the principles of scientific rigor, quantifiability, and systematic logic. It is grounded in internationally recognized evaluation systems, including the UN World Tourism Organization (UNWTO) performance domains and the World Economic Forum (WEF) Travel & Tourism Competitiveness Index (TTCI), as well as policy documents and evaluation norms issued by China's Ministry of Culture and Tourism (MCT). The indicators have been refined to reflect the local context of the YRDCC, aligning global benchmarking with region-specific development practices. Each of the 20 indicators was selected based on three criteria: 1) Theoretical relevance to brand tourism competitiveness, 2) Empirical accessibility based on reliable multi-year statistical records (2019 – 2023), and 3) Data comparability and public transparency across all 27 cities in the YRDCC.

TABLE I. INDICATORS OF EVALUATION FACTORS OF BRAND TOURISM COMPETITIVENESS

Target Level	Criteria	Indicator Level	Unit	Indicator Direction
A Dynamic Assessment of Branded Tourism Competitiveness in the Yangtze River Delta City Cluster Agglomerations	B1 Comprehensive strength of the city	C1 GDP per capita	billion	+
		C2 Regional GDP	billion	+
		C3 Share of tertiary GDP	percent	+
		C4 Proportion of employment in the tertiary sector	percent	+
		C5 Total retail sales of consumer goods	billion	+
	B2 Brand Tourism Development Scale	C6 Domestic Tourism	million	+
		C7 Number of international tourists	million	+
		C8 Domestic branded tourism revenue	billion	+
		C9 Foreign exchange earnings from branded tourism	billion	+
	B3 Brand tourism resources	C10 Natural landscape resources	one	+
		C11 Number of historical and cultural sites	one	+
		C12 Branded Event Resources	one	+
		C13 Word-of-mouth evaluation of city tourism brands by tourists and the public	Point scale	+
	B4 Brand tourism service level	C14 Number of star-rated hotels	one	+
		C15 Number of tourist agencies	one	+
		C16 Number of public restrooms	one	+
		C17 Number of museums	one	+
		C18 Cultural and brand communication organizations	one	+
	B5 Urban environmental quality	C19 Green space coverage rate of built-up areas	pieces	+
		C20 Green space per capita	m^2/person	+

The majority of the data were obtained from government-authorized sources, primarily national and provincial statistical yearbooks. Statistical Yearbooks: China City Statistical Yearbook, China Urban Construction Statistical Yearbook, Provincial and municipal yearbooks (e.g. Shanghai Statistical Yearbook, Anhui Statistical Yearbook). Official Bulletins and Thematic Reports, China Culture and Tourism Development Statistical Bulletin (Ministry of Culture and Tourism, 2019 – 2023) Annual Work Reports from local Bureaus of Culture and Tourism across Jiangsu, Zhejiang, and Anhui. Public

Databases and API Platforms: National Data Platform (data.stats.gov.cn), Online travel platforms (Ctrip, Dianping, Mafengwo) for indicators such as C13 Visitor Reviews, using NLP-based sentiment analysis and expert-weighted scoring. Mapping and POI services (Gaode Map, Baidu Map API) for facility density and infrastructure coverage indicators.

Based on the multidimensional structure of urban brand tourism competitiveness, this study establishes an evaluation framework comprising five key criteria: urban comprehensive strength, scale of brand tourism development, brand tourism resources, quality of brand tourism services, and urban environmental quality, thereby forming a systematic indicator system.B1 Urban Comprehensive Strength reflects a city's economic foundation and development potential, serving as a fundamental support for building its tourism brand. This criterion includes five indicators (C1 – C5): per capita GDP, total regional GDP, the proportion of the tertiary industry, employment in the tertiary sector, and total retail sales of consumer goods. These indicators evaluate the city's overall development level in terms of economic output, industrial structure, and consumer vitality. B2 Scale of Brand Tourism Development represents the degree of tourism development and market attractiveness. Indicators C6 to C9 measure the number of domestic and international tourists received and related tourism revenues, providing a direct reflection of the size of the tourism market and its economic contribution. B3 Brand Tourism Resources focus on assessing the city's natural endowments and cultural characteristics. Indicators C10 to C13 include natural landscapes, historical and cultural heritage sites, branded event resources, and visitor reviews — collectively capturing the city's tourism appeal and brand perception. B4 Quality of Brand Tourism Services emphasizes the supply capacity and experiential quality of tourism services. Indicators C14 to C18 include the number of star-rated hotels, travel agencies, public restrooms, museums, and cultural or brand communication institutions, reflecting the city's capabilities in service provision and cultural promotion. B5 Urban Environmental Quality is assessed using indicators C19 and C20: the green coverage rate in built-up areas and per capita green space. These indicators evaluate the city's ecological livability and capacity for sustainable development.

This study further proposes a five-dimensional interactive model—Economy – Resources – Services – Communication – Environment — highlighting the systematic coupling and dynamic synergy among various indicators. Urban economic strength supports the development of infrastructure and service systems, providing the material foundation for the expansion of brand tourism. Rich tourism resources are the core attraction for visitors, but their value realization depends on the service system's carrying capacity and management efficiency. The quality of tourism services and the effectiveness of digital communication directly influence visitor satisfaction and brand reputation, fostering brand loyalty and enhancing urban competitiveness. A high-quality ecological environment not only improves the visitor experience but also ensures the sustainability of brand tourism. Meanwhile, digital content dissemination and feedback mechanisms based on tourist behavior further promote the visibility and communication efficiency of the city brand.

III. Results

A. Analysis of Factors of City Brand Tourism Competitiveness

B1 — Urban Comprehensive Strength: This dimension reflects a city's economic foundation and its capacity to support tourism infrastructure, service provision, and brand promotion — factors that are critical to enhancing tourism competitiveness. According to the data, Shanghai ranks first with an overwhelming score of 0.705, significantly ahead of other cities. Wenzhou (0.511) and Suzhou (0.503) follow in second and third place, respectively, indicating that robust economic development provides a strong foundation for tourism competitiveness. Although Nanjing (0.463) and Hangzhou (0.483) score slightly lower, their relatively balanced performance suggests solid economic bases and well-developed service sectors. In contrast, cities such as Chuzhou (0.046) and Tongling (0.051) remain at the bottom, reflecting limitations in resource allocation and industrial structure (Fig. 2). Overall, the distribution of scores in this dimension reveals a distinct pattern: strong concentration at the top, moderate clustering in the middle, and dispersion at the lower end.

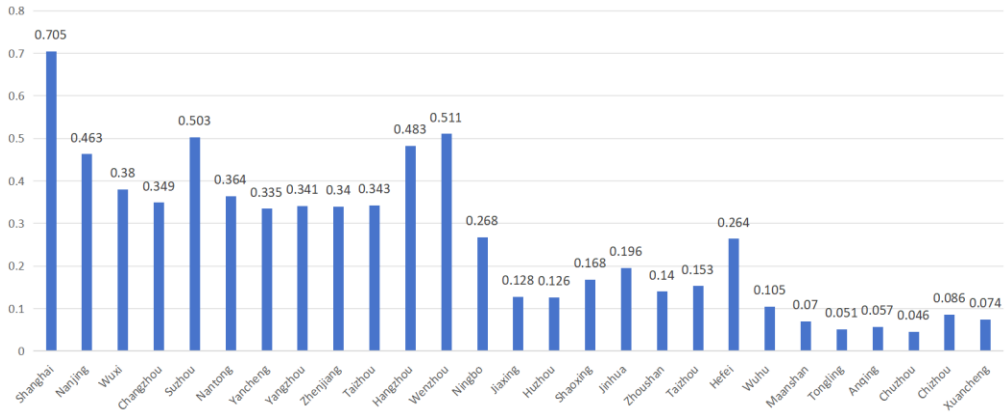


Fig. 2. B1 score distribution.

B2 — Scale of Branded Tourism Development: This dimension evaluates the external expansion capacity of the tourism market. Shanghai (1.000) significantly outperforms all other cities, demonstrating a dominant market scale and a strong brand-driven "magnetic effect" that attracts tourists across the region. Suzhou (0.578) ranks second, reflecting robust tourism reception capacity and efficient conversion of tourism flows into economic returns. Cities in the mid-range, such as Nanjing (0.226) and Wuxi (0.168), exhibit steady growth in both visitor numbers and tourism revenue, indicating solid development potential. In contrast, cities like Yancheng (0.015) and Taizhou (0.019) show limited tourism activity and weak foundational support. Overall, the distribution of scores in this dimension reveals a pronounced polarization: top-tier cities benefit from well-developed tourism ecosystems, while lower-tier cities remain in the early stages of development (Fig. 3). These cities must draw lessons from leading counterparts, particularly in enhancing the quality of tourism offerings and improving service capabilities.

B3—Branded Tourism Resources: This dimension reflects a city's tourism appeal and its cultural communication foundation. Notably, Yancheng (0.560), Hefei (0.445), and Wuhu (0.404) rank among the top three, indicating that certain cities in Anhui Province have demonstrated strong performance in integrating natural and cultural resources. For instance, Hefei benefits from the promotion of Binhu cultural assets and branded events, while Wuhu effectively leverages its rich historical heritage. Conversely, cities such as Zhoushan (0.072) and Jiaxing (0.080), despite possessing considerable resource endowments, exhibit underdeveloped capabilities in brand development and public communication. The overall distribution of scores in this dimension is relatively balanced (primarily within the 0.1 – 0.4 range), suggesting that while most cities hold comparable levels of tourism resources, their ability to transform these assets into recognizable and marketable tourism brands varies significantly (Fig. 4).

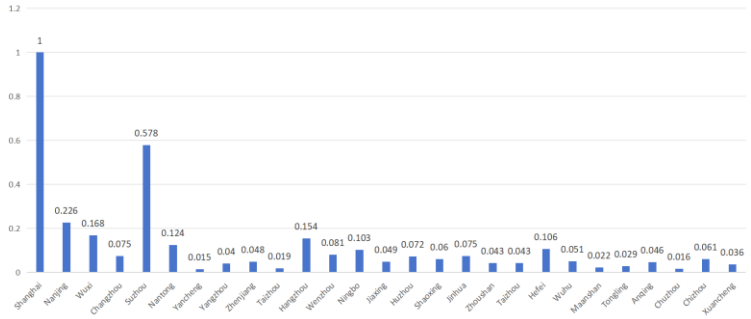


Fig. 3. B2 score distribution.

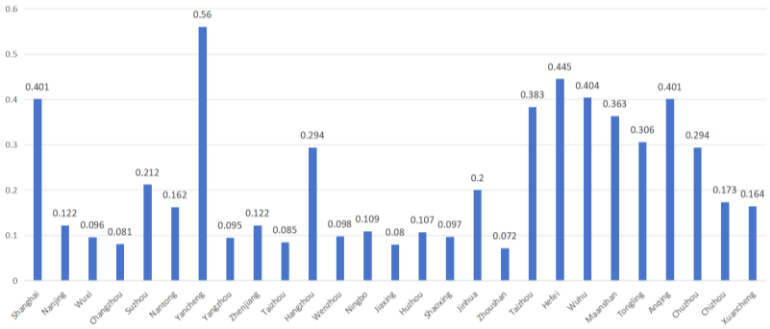


Fig. 4. B3 score distribution.

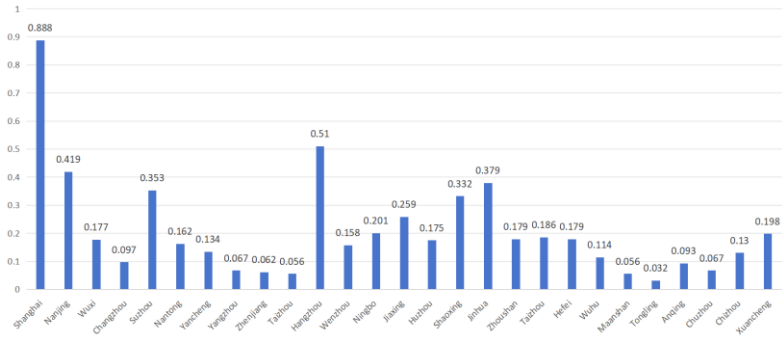


Fig. 5. B4 score distribution.

B4—Branded Tourism Service Level: Service capacity constitutes the perceptual foundation of a city's tourism competitiveness. The data show that Shanghai (0.888) ranks first, with significant advantages in high-end hotel availability, cultural and tourism facilities, and brand communication institutions. Hangzhou (0.510) and Nanjing (0.419) follow closely, indicating relatively well-developed service systems. In contrast, cities such as Tongling (0.032), Chuzhou (0.067), and Zhenjiang (0.062) exhibit notably lower scores, reflecting outdated infrastructure and limited tourism reception capacity (Fig. 5). The wide score range in this dimension highlights pronounced disparities among cities and demonstrates that investment in tourism services directly influences public perception and word-of-mouth dissemination of the city's brand.

B5—Urban Environmental Quality: Ecological livability has become increasingly intertwined with the sustainable development of tourism. Cities, such as Chizhou (0.819), Huzhou (0.764), and Yangzhou (0.752) lead in this dimension, reflecting outstanding performance in green space coverage and ecological preservation. In contrast, major urban centers like Shanghai (0.000) and Wenzhou (0.191) score significantly lower, largely due to high urbanization levels that constrain ecological space (Fig. 6). Scores in this dimension are broadly distributed between 0.2 and 0.8, indicating a relatively balanced performance overall. This suggests that in several cities, ecological civilization efforts have effectively translated into tourism advantages.

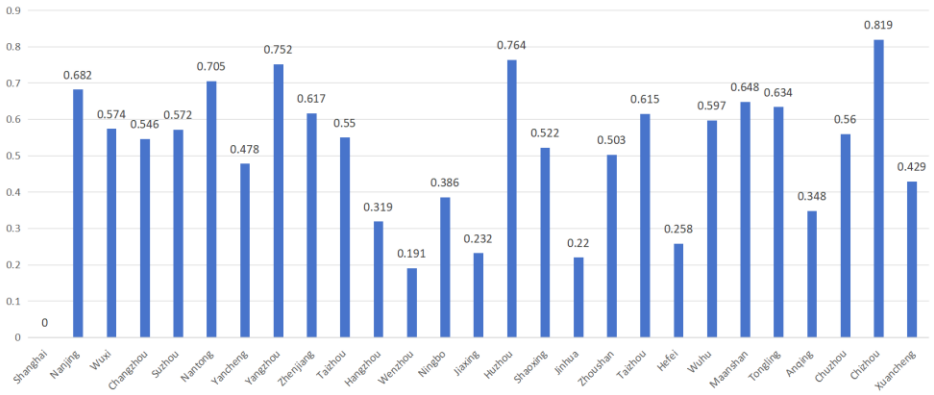


Fig. 6. B5 score distribution.

This study develops a five-dimensional interactive model—Economy – Resources – Services – Communication – Environment—to highlight the systemic synergy among the core components of brand tourism competitiveness. Specifically, urban comprehensive strength (B1) provides the financial and policy foundation for tourism development; resource endowment (B3) constitutes the fundamental appeal to tourists; service quality (B4) shapes visitor experience and word-of-mouth dissemination; development scale (B2) reflects the market transformation capacity of tourism; and environmental quality (B5) underpins the long-term sustainability of tourism systems. In practical application, high-performing cities such as Shanghai and Hangzhou can leverage their institutional resources and brand advantages to establish regional "flagship demonstration zones"—offering replicable models in areas such as infrastructure development, brand communication, and green tourism governance. In contrast, lower-performing cities like Chuzhou and Tongling may benefit from adopting mechanisms such as cross-regional tourism passes, shared resource platforms, and joint marketing strategies, thereby enhancing brand visibility and improving tourism service capacity through collaborative growth. To improve the comparability and validity of soft indicators, particularly the tourist sentiment index (C13), this study integrates social media data crawling techniques with expert scoring mechanisms, enabling data standardization and objectivity Supporting Information (S1 Table). Additionally, Gaode Map and Baidu Map APIs are employed to enhance the spatial granularity in the measurement of tourism service infrastructure, thus improving the analytical precision of the service dimension. Overall, the brand tourism competitiveness of the YRDCC exhibits pronounced characteristics of multidimensionality, imbalance, and regional differentiation. The proposed indicator system not only uncovers structural disparities across cities in terms of competitiveness factors but also offers a practical framework for future policy optimization and regional coordination. Subsequent research may further integrate visitor movement patterns, smart tourism platform data, and other real-time dynamic indicators to construct a more adaptive and time-sensitive evaluation system for urban tourism competitiveness.

B. Synthesize and Analyze

This study employs the entropy-weighted TOPSIS method to dynamically evaluate the overall performance of brand tourism competitiveness across 27 cities in the YRDCC from 2019 to 2023. The relative closeness to the ideal solution is used as a metric to assess how closely each city's tourism performance aligns with the optimal scenario-the closer the value is to 1, the stronger the competitiveness. Calculate the relative proximity of each city through the formula $c_i = \frac{D_i^-}{D_i^+ + D_i^-}$, where, D_i^+ is the distance of the city from the ideal solution and D_i^- is the distance of the city from the negative ideal solution. The evaluation process involves the following steps: First, the optimal (maximum) and anti-optimal (minimum)

values for each indicator are identified. Second, the Euclidean distances between each city and both the ideal and anti-ideal solutions are calculated. Finally, each city's comprehensive score is derived based on the relative closeness formula. Furthermore, ArcGIS 10.8.1 is employed to visualize the spatial distribution of competitiveness levels (Fig. 7), thereby revealing the spatiotemporal evolution [33] of brand tourism competitiveness over the study period.

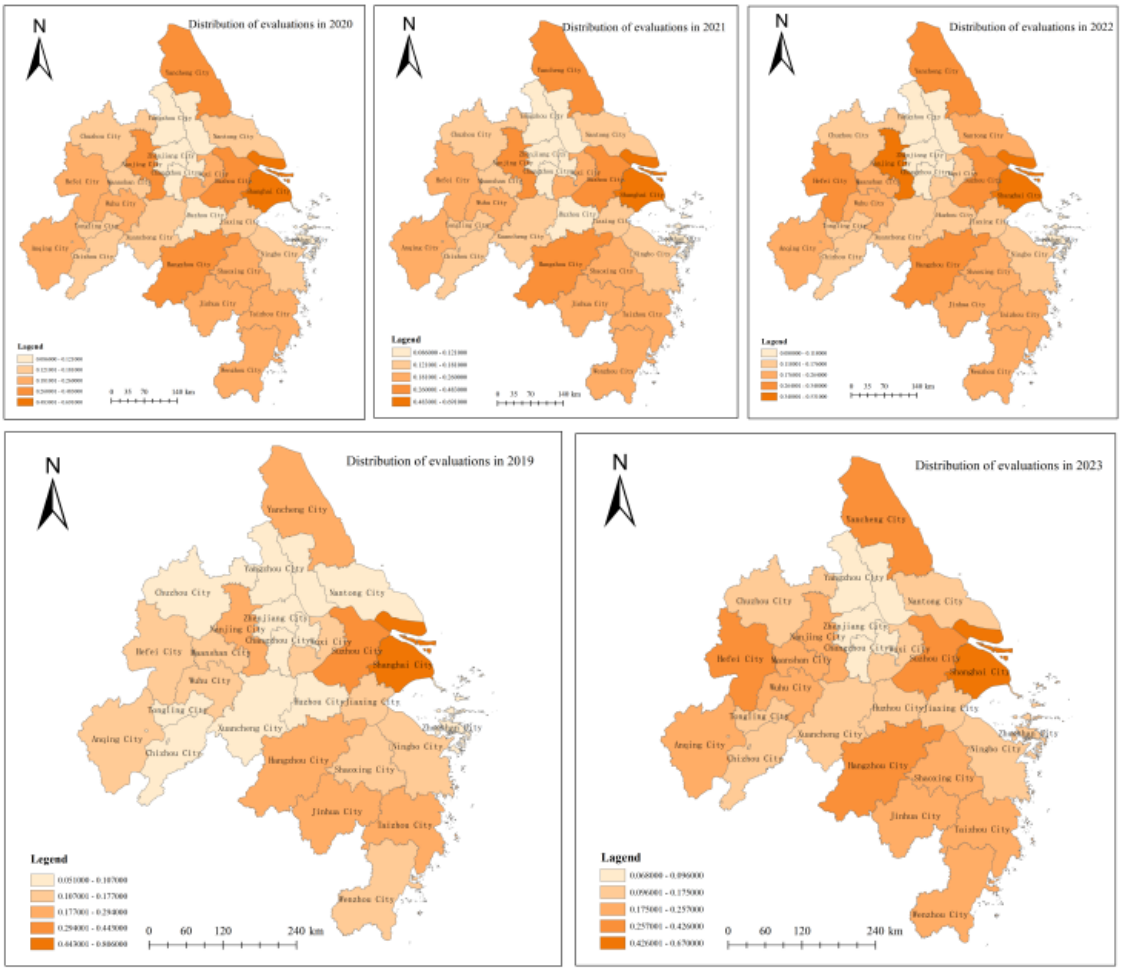


Fig. 7. Evaluation Distribution of the YRDCC.

High-performing and stable cities such as Shanghai, Suzhou, and Hangzhou have consistently maintained strong brand tourism competitiveness (Table II). From 2019 to 2023, Shanghai remained in the top position, with its relative closeness score ranging from 0.892 to 0.901. This reflects its comprehensive advantages in economic strength, international brand development, well-developed service infrastructure, and the effective integration of cultural resources, positioning it as a leading driver in regional brand tourism. Suzhou (0.601 – 0.624) and Hangzhou (rising from 0.538 to 0.689) demonstrated steady and coordinated development between brand cultivation and resource utilization, with continued efforts particularly in digital tourism, cultural and creative industries, and the provision of high-end tourism services. Cities with steadily rising performance, such as Hefei, Nantong, and Ningbo, also show promising momentum. Hefei advanced from 8th place in 2019 to 6th in 2023, with a 30.3% increase in its relative closeness score, indicating continuous progress in tourism infrastructure, investment in cultural tourism projects, and integrated brand communication. Nantong and Ningbo have achieved slow but steady gains by optimizing resource allocation and enhancing service quality. In contrast, cities showing significant decline, such as Zhenjiang and Yangzhou, have seen their rankings drop notably. Zhenjiang fell from 7th in 2019 to 12th in 2023 (a decline of 11.4%), while Yangzhou dropped from 6th to 11th (a decline of 13.1%). These setbacks are primarily attributed to lagging brand marketing, low resource utilization, and insufficient transportation connectivity. This suggests that small and medium-sized cities face the risk of marginalization if tourism strategies are not timely adjusted. The data reveal a clear trend of polarization and structural imbalance. The distribution of relative closeness scores exhibits a "top-heavy, sparse middle, scattered tail" pattern. For instance, in 2023, the gap between Shanghai (0.901) and Hefei (0.615) was considerable, while most mid- and lower-ranking cities clustered between 0.3 and 0.45. This indicates persistent disparities in critical factors such as infrastructure, brand maturity, and ecological support across the region.

TABLE II. BRANDED TOURISM RESPONSE FOR THE YRDCC, 2019-2023.

Items	D_i^+	D_i^-	c_i	Ranking
R1	0.72	0.77	0.79	1
R2	1.51	1.77	2.15	5
R3	3.87	4.8	5.96	15
R4	6.08	7.57	9.44	24
R5	0.8	0.92	1.04	2
R6	3.65	4.49	5.58	14
R7	1.24	1.47	1.76	4
R8	6.56	8.18	10.21	26
R9	6.32	7.87	9.82	25
R10	6.81	8.49	10.59	27
R11	1	1.17	1.37	3
R12	3.15	3.89	4.81	12
R13	4.11	5.1	6.34	16
R14	4.36	5.42	6.73	17
R15	5.61	6.97	8.68	22
R16	2.9	3.57	4.42	11
R17	1.93	2.36	2.88	7
R18	5.84	7.27	9.06	23
R19	2.17	2.65	3.26	8
R20	1.71	2.07	2.51	6
R21	2.42	2.97	3.65	9
R22	3.39	4.19	5.19	13
R23	4.62	5.74	7.13	18
R24	2.66	3.28	4.03	10
R25	4.87	6.04	7.51	19
R26	5.35	6.66	8.28	21
R27	5.11	6.35	7.9	20

High-performing cities should further extend the reach of their brand communication by developing high-end international tourism products and globally influential cultural IPs. They are also encouraged to continue serving as hubs of innovation and leadership, radiating positive effects to surrounding cities and driving regional integration. Mid-tier cities should focus on deeply exploring their unique cultural characteristics and establishing differentiated market positioning. Strengthening mechanisms for inter-city cooperation and regional coordination is essential to enhance the overall brand value chain and achieve synergistic development. Lower-performing cities need to adopt targeted improvement strategies. These include enhancing transportation connectivity, diversifying tourism product offerings, and encouraging the entry of private capital into the cultural and tourism sectors. Additionally, they can learn from the best practices of high-performing cities through "contextualized transplantation" and joint promotional efforts, thereby accelerating their competitiveness in a more customized and collaborative manner.

IV. DISCUSSION

A. Theoretical and Practical Significance

Theoretical significance: First, this study expands the multidimensional measurement framework of urban brand development. While traditional research on city branding has primarily focused on subjective dimensions such as perceived identity and image construction, this study introduces objective indicators — including resources, services, environment, and economy — into a systematic model. By filling the theoretical gap in the quantifiable assessment of city brands, the research contributes to shifting the focus of brand tourism from impression-based narratives to comprehensive competitiveness evaluation. Second, the study enriches the methodological foundation for assessing regional tourism synergy and competitiveness. By employing the TOPSIS model in combination with the entropy weight method, this research is among the first to systematically evaluate inter-city differences and structural characteristics of brand tourism competitiveness at the scale of Chinese urban agglomerations. It reveals the weighting mechanisms of key success factors — such as resource transformation, service provision, and ecological capacity — in the process of brand development, thereby enhancing the explanatory power and practical applicability of the theoretical model. Third, the study responds to the theoretical proposition of "structural disparities and collaborative governance" in urban tourism competitiveness. By empirically analyzing the performance of cities across five dimensions (B1 – B5) and their complementary relationships, this research concretizes the concepts of "coordinated development" and "complementary advantages". For instance, core cities such as Shanghai can foster brand enhancement in peripheral cities through infrastructure spillovers and institutional connectivity. Conversely, lower-tier cities may reduce market entry barriers via collaborative resource allocation and coordinated marketing, jointly advancing the formation of a multi-layered and interactive regional brand competitiveness framework.

Relevance: First, the study supports China's national strategy for promoting coordinated regional development. As brand tourism is one of the core pillars of YRDCC integration, the findings help accurately identify the relative position and weaknesses of each city within the brand competitiveness system. This provides a valuable data foundation for the coordinated allocation of cultural and tourism resources, cross-regional integration, and targeted policy implementation. Second, it contributes to enhancing the international communication capacity and cultural influence of urban agglomerations. In the context of globalization, brand tourism serves as a key vehicle for cities to participate in international discourse and cultural exchange [50]. The study highlights how cities such as Hangzhou and Hefei have elevated their brand levels through smart tourism platforms and international events, illustrating the critical role of branding in embedding cities into global tourism networks. Third, the research facilitates regional complementarity and structural optimization. By uncovering structural disparities among cities across dimensions such as resources, services, and the environment, it proposes replicable branding strategiesz, such

as “flagship-driven development”, “differentiated positioning”, and “ecological synergy”, offering theoretical support for constructing a tourism development model based on “differentiated positioning + coordinated integration”.

B. Competitiveness Analysis and Policy Recommendations

This study employs the entropy-weighted TOPSIS model to construct a comprehensive evaluation system comprising five criterion layers and 20 core indicators, enabling an effective assessment of the hierarchical characteristics of brand tourism competitiveness across the YRDCC. The findings reveal significant disparities among cities in terms of economic foundations, resource endowments, service infrastructure, and ecological environment—factors that fundamentally drive the differentiation in tourism competitiveness.

Cities such as Shanghai and Hangzhou have consistently ranked among the top in tourism competitiveness, owing to their strong economic foundations, high tourism reception capacity, and significant international influence. However, they also face challenges such as sluggish brand renewal and mounting ecological pressures. It is recommended that these cities further enhance their leadership by developing "regional tourism flagship demonstration zones" and strengthening resource sharing and brand collaboration with neighboring cities — through initiatives such as the YRDCC Cultural Tourism Expo and joint city marketing platforms. Additionally, adopting green tourism policies, including low-carbon transportation and carbon-neutral scenic areas, can help alleviate ecological constraints and sustain their global competitiveness.

Cities such as Suzhou and Ningbo possess rich cultural resources and robust manufacturing bases, but their levels of brand recognition remain moderate. However, they still face limitations in the provision of high-end services and international outreach. To bridge these gaps, targeted efforts should be made to develop distinctive cultural IPs (intellectual properties) that highlight local heritage and creativity. For example, Suzhou could deepen the digital presentation of its classical garden culture, while Ningbo might highlight its “Maritime Silk Road” heritage and expand port-related tourism development. Furthermore, enhancing cooperation with international tourism organizations, such as introducing multilingual guided services and launching overseas advertising campaigns, can significantly improve both the appeal and service quality for international visitors [51].

Cities such as Yangzhou, Zhenjiang, and Taizhou face multiple challenges, including low brand visibility, underdeveloped infrastructure, and weak digital services. To enhance their competitiveness, priority should be given to improving transportation accessibility and tourism reception capacity through the development of high-grade hotels and visitor centers. Drawing on the successful experiences of nearby high-performing cities, these cities could adopt regional travel passes and integrated transportation systems to promote tourist mobility and resource complementarity. Additionally, efforts should be made to strengthen local festival brands, for example, Yangzhou could leverage the Slender West Lake Cultural Festival as a stable platform to attract visitors and enhance brand recognition.

It is recommended that the entire region implement an integrated green tourism development strategy. Small and medium-sized cities can enhance their ecological brand value by jointly developing ecological corridors and launching collaborative low-carbon tourism routes. The promotion of green transportation option, such as electric vehicles and public bicycles, alongside the establishment of smart management systems will enable real-time monitoring of tourism resources and environmental carrying capacity. These measures will collectively strengthen the region’s capacity for sustainable tourism competitiveness.

C. Domestic and International Case Studies and Lessons

Enhancing tourism competitiveness requires more than abundant resource endowments; it hinges on the synergistic interaction of institutional innovation, cultural identity, and environmental sustainability [49]. The following representative cases offer valuable lessons for the YRDCC in its pursuit of high-quality and differentiated tourism development.

Kyoto, Japan-Cultural Continuity through Festival Branding, Kyoto exemplifies how traditional cultural assets can be embedded into everyday urban life and tourism practice. Through the preservation of historical architecture and the regular hosting of festivals such as the Gion Matsuri, Kyoto has institutionalized cultural rituals that enhance visitor loyalty and deepen place-based identity. For cities in the YRDCC, this suggests the value of a “cultural embedding” strategy — transforming intangible cultural heritage, such as Jiangnan water-town traditions, into dynamic experiences via immersive tourism formats like themed performances, cultural fairs, and participatory heritage events.

Florence, Italy-Balancing Heritage Conservation with Tourism Development, Florence demonstrates how cultural heritage and tourism can coexist through well-designed institutional frameworks. Mechanisms such as “cultural usage rights” and designated “historic architecture control zones” enable the city to maintain cultural authenticity while promoting adaptive reuse. These practices offer policy implications for YRDCC cities, particularly Suzhou and Anhui’s Huizhou villages, which face similar pressures to reconcile heritage conservation with increasing tourism demands. Drawing on Florence’s approach could support the development of sustainable, heritage-based tourism strategies across the YRDCC.

Freiburg, Germany – Advancing Green Transformation in Urban Tourism, Freiburg’s reputation as a “green city” stems from its successful integration of environmental goals into tourism planning. Through the development of low-carbon scenic zones, comprehensive public transit networks, and environmentally certified accommodations, Freiburg exemplifies ecologically responsible tourism. Given the environmental constraints faced by the densely populated urban cores of the YRDCC, this case underscores the need to formulate regional “low-carbon tourism development guidelines,” implement ecological carrying capacity assessments in popular destinations, and encourage responsible visitor behavior through policy incentives and public education.

Pearl River Delta (PRD) vs. YRDCC: Divergent Pathways of Tourism Competitiveness. A domestic comparative analysis between the PRD and YRDCC highlights their divergent strategic orientations in tourism development. The PRD, encompassing cities such as Guangzhou, Shenzhen, and Zhuhai, has pursued a market-oriented, globally integrated model, leveraging mechanisms such as free trade zones, port infrastructure, and large-scale commercial events (e.g. Canton Fair, Shenzhen Tech Expo) to drive tourism growth. This model prioritizes volume, innovation, and modern leisure consumption.

By contrast, the YRDCC adopts a development logic centered on cultural preservation, ecological balance, and regional coordination. Cities like Hangzhou and Suzhou have cultivated tourism brands rooted in Jiangnan cultural identity, integrating heritage conservation with high-quality visitor experiences. While the PRD excels in infrastructure openness and digital marketing, it faces challenges in fostering cohesive cultural narratives and safeguarding intangible heritage. The YRDCC, though more conservative in its pace of innovation, offers a more sustainable and culturally enriched model through its emphasis on regional synergy and ecological tourism. This comparison suggests that the YRDCC could benefit from selectively adopting the PRD's strengths in digitalization and international outreach, while offering in return a replicable model of integrated, heritage-centered, and ecologically sustainable tourism development. Such a synthesis would allow both regions to enhance their competitiveness while aligning with global trends toward responsible and differentiated tourism branding.

D. Research Limitations and Future

This study has carried out in-depth exploration in the construction of the indicator system and assessment methods, but the following limitations still exist. First, data acquisition limitations, part of the city tourism-related data is not detailed enough, the future should be introduced to improve data collection by means of big data monitoring, tourist satisfaction surveys and other means. Second, dynamic monitoring is insufficient, failing to fully reveal the long-term impact of policy changes and other external factors on tourism, it is recommended to combine with the panel data model to dynamically monitor the evolution mechanism of competitiveness in the future. Third, the integration of multi-source data, the current assessment mainly relies on macro statistics, in the future, social media data, tourists' word-of-mouth evaluation and other information sources can be introduced to enrich the research dimension. Fourth, smart tourism and digital development, in the future, attention should be paid to the empowerment of smart tourism platforms and digital management tools on the competitiveness of city brands, and explore the role of digital technology in promoting the upgrading of the cultural and tourism industry.

V. CONCLUSIONS

This study focuses on the YRDCC and investigates the structural measurement and dynamic comparison of regional brand tourism competitiveness. A comprehensive indicator system was constructed, encompassing five dimensions: overall urban strength, scale of brand tourism development, tourism resources, service quality, and environmental conditions. The framework includes 20 secondary indicators, with clearly defined data sources and a scientifically grounded structure. It aligns closely with the World Economic Forum's framework for tourism competitiveness while incorporating region-specific factors such as digital communication and sustainable development. Using the entropy-weighted TOPSIS method, the study applies objective weighting to calculate the relative closeness of each city to an ideal solution. This enables a comprehensive evaluation of the overall performance, structural characteristics, and spatial evolution of brand tourism competitiveness among YRDCC during the period from 2019 to 2023.

The competitive landscape of brand tourism in the YRDCC region exhibits a spatial pattern characterized by "dominant leadership and distinct gradients." Cities like Shanghai and Hangzhou have consistently ranked at the top in terms of competitiveness, demonstrating strong capacities in resource integration, economic support, and brand diffusion. These cities are typical examples of "lagship cities". Meanwhile, cities such as Suzhou, Nanjing, and Ningbo maintain relatively stable performance in terms of cultural resources and tourist reception, yet still have room for improvement in global brand development and cultural IP export. At the lower end of the spectrum, cities like Chuzhou, Tongling, and Zhoushan struggle with underdeveloped infrastructure, weak digital services, and limited brand communication capacity, highlighting the ongoing issue of regional development imbalance. The indicators used in this study exhibit interrelated and synergistic effects. The proposed "Five-Dimensional Synergy" model reveals that: B1 (Overall Urban Strength) provides essential capital and policy support for tourism development; B2 (Development Scale) reflects market conversion and external expansion capacity; B3 (Resource Base) determines core attractiveness; B4 (Service Quality) influences tourist experience and word-of-mouth dissemination; B5 (Environmental Quality) serves as the foundation for building green tourism brands. These dimensions are highly interlinked across different cities, collectively shaping their brand tourism competitiveness structures. In terms of temporal evolution, cities such as Hefei and Hangzhou have shown notable upward momentum, each climbing two ranks, indicating that investments in smart tourism, infrastructure, and brand-related events have yielded significant results. In contrast, cities like Zhenjiang and Yangzhou have experienced a marked decline in competitiveness, reflecting delays in cultural-tourism integration, market responsiveness, and product innovation. Cities such as Shanghai and Suzhou continue to demonstrate stable and long-term competitive advantages, highlighting their strong resilience within the regional brand tourism system. From a spatial perspective, GIS-based visualization results reveal a "core – periphery" distribution pattern. High-competitiveness cities are mainly concentrated in the Shanghai – Jiangsu – Zhejiang corridor, whereas most cities in Anhui remain in developmental lowlands. Although cities like Chizhou and Wuhu possess ecological advantages, these have not yet been effectively transformed into brand strengths, underscoring the urgent need to enhance regional resource coordination mechanisms.

This study makes three key contributions at both theoretical and practical levels: First, it expands the theoretical framework for evaluating urban brand competitiveness, particularly by integrating heterogeneous indicators from multiple sources and combining objective and subjective data analysis. This approach addresses the existing gap in comprehensive understanding of city brand tourism in the current literature. Second, by constructing a five-dimensional interactive framework encompassing economy, resources, services, communication, and environment, the study strengthens the causal relationships and synergetic logic among indicators. This provides a more structured and coherent approach to modeling brand tourism competitiveness. Third, methodologically, the integration of the entropy-weighting method and the TOPSIS ranking model overcomes the limitations of traditional subjective weighting approaches. It enhances both the scientific rigor and operational feasibility of cross-city comparative analysis.

High-ranking cities such as Shanghai and Hangzhou should play a leading role by establishing regional "flagship demonstration zones". These cities can spearhead collaborative brand-building initiatives, including co-hosted festivals, inter-city tourism alliances, and shared platform mechanisms, to drive coordinated upgrades among mid- and low-ranking cities. Mid-level cities like Suzhou, Ningbo, and Hefei should focus on leveraging their unique resources and cultural symbols to strengthen differentiated brand positioning. Efforts should also be made to enhance international visibility and brand recognition. Lower-tier cities, including Chuzhou, Tongling, and Zhenjiang, should prioritize addressing infrastructure gaps and improving digital service capabilities. Through targeted policy support, strategic market partnerships, and in-depth cultural resource development, these cities can "scale up by leveraging external strengths". At the regional level, a green and low-carbon transformation should be actively pursued. By integrating ecological preservation with brand development, the YRDCC region can establish a model that combines "ecological space + smart management + green communication", thereby enhancing long-term sustainable competitiveness.

ACKNOWLEDGMENT

We thank the reviewers and the editor for helpful comments on this manuscript.

AUTHORS' CONTRIBUTIONS

Conceptualization: Daoyi Wu.

Data curation: Dongmei Wang.

Funding acquisition: Daoyi Wu.

Investigation: Daoyi Wu, Dongmei Wang.

Methodology: Daoyi Wu, Dongmei Wang.

Software: Daoyi Wu, Dongmei Wang.

Visualization: Daoyi Wu, Dongmei Wang.

Writing – Original draft: Dongmei Wang.

Writing – Review and editing: Daoyi Wu, Dongmei Wang.

FUNDING

This research was supported by the Anhui Cultural Tourism Innovative Development Research Institute 2024 Director's Fund Project (Project No. ACTK2024ZD01), "Philosophy and Social Science Research Programs for Universities in Anhui Provinc" (Project No. 2024AH053112), "Adolescent Mental Health and Crisis Intelligent Intervention Open Fund Project of Key Laboratory of Philosophy and Social Science of Anhui Province" (Project No.SYS2024B16).

INFORMED CONSENT STATEMENT

All participants in this study provided their informed consent.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors on request.

CONFLICTS OF INTEREST

The authors have no conflicts of interest to declare.

REFERENCES

[1] Martínez-González, J.A.; Díaz-Padilla, V.T.; Parra-López, E. Study of the Tourism Competitiveness Model of the World Economic Forum Using Rasch's Mathematical Model: The Case of Portugal. Sustainability 2021, 13, 7169. https://doi.org/10.3390/su13137169

[2] Garanti, Z.; Ilkhanizadeh, S.; Liasidou, S. Sustainable Place Branding and Visitors' Responses: A Systematic Literature Review. Sustainability 2024, 16, 3312. https://doi.org/10.3390/su16083312.

[3] Camprubí, R.; Gassiot-Melian, A. Advances in Tourism Image and Branding. Sustainability 2023, 15, 3688. https://doi.org/10.3390/su15043688.

[4] Jiang, F.; Huang, R.; Chen, Q.; Zhang, J. Brand Equity, Tourist Satisfaction and Travel Intentions in a UNESCO Creative City of Gastronomy: A Case Study of Yangzhou, China. Foods 2023, 12, 2690. https://doi.org/10.3390/foods12142690

[5] Zhang, S.; Chi, L.; Zhang, T.; Wang, Y. Optimization of Tourism Management Based on Regional Tourism Competitiveness Evaluation: Evidence from Ningxia Hui Autonomous Region, China. Sustainability 2023, 15, 9591. https://doi.org/10.3390/su15129591

[6] Alamrawy, M.A.T.; Hassan, T.H.; Saleh, M.I.; Abdelmoaty, M.A.; Salem, A.E.; Mahmoud, H.M.E.; Abdou, A.H.; Helal, M.Y.; Abdellmonaem, A.H.; El-Sisi, S.A.-W. Tourist Attribution toward Destination Brands: What Do We Know? What We Do Not Know? Where Should We Be Heading? Sustainability 2023, 15, 4448. https://doi.org/10.3390/su15054448

[7] Wang, F.; Zhang, H.; Zhou, J. Impact of Green Finance on Chinese Urban Land Green Use Efficiency: An Empirical Study Based on a Quasinatural Experiment. Land 2025, 14, 332. https://doi.org/10.3390/land14020332

[8] Wang, P.; Li, X.; Zhang, L.; Wang, Z.; Bai, J.; Song, Y.; Han, H.; Zhao, T.; Huang, G.; Yan, J. Spatiotemporal Variations of Production–Living–Ecological Space under Various, Changing Climate and Land Use Scenarios in the Upper Reaches of Hanjiang River Basin, China. Land 2023, 12, 1770. https://doi.org/10.3390/land12091770

[9] Zhou, Z.; Liu, Z.; Wang, G. Driving Sustainable Cultural Heritage Tourism in China through Heritage Building Information Modeling. Buildings 2024, 14, 3120. https://doi.org/10.3390/buildings14103120.

[10] Xu, H.; Xu, N. Industrial Co-Agglomeration and Urban Green Total Factor Productivity: Multidimensional Mechanism and Spatial Effect. Sustainability 2024, 16, 9415. https://doi.org/10.3390/su16219415

[11] Ren, M.; Chai, N. Resilience Renewal Design Strategy for Aging Communities in Traditional Historical and Cultural Districts: Reflections on the Practice of the Sizhou'an Community in China. Buildings 2025, 15, 965. https://doi.org/10.3390/buildings15060965

[12] Yu, Z.; Meng, X.; Yu, G. Evolution of "Production–Living–Ecological" Spaces Conflicts and Their Impacts on Ecosystem Service Values in the Farming–Pastoral Ecotone in Inner Mongolia During Rapid Urbanization. Land 2025, 14, 447. https://doi.org/10.3390/land14030447

[13] Li, T.; Wang, X.; Jia, H. Evaluate Water Yield and Soil Conservation and Their Environmental Gradient Effects in Fujian Province in South China Based on InVEST and Geodetector Models. Water 2025, 17, 230. https://doi.org/10.3390/w17020230

[14] Jing, X.; He, Y.; Sun, Y.; Wang, M.; Wang, X. Spatial–Temporal Divergence and Coupling Analysis of Land Use Change and Ecosystem Service Value in the Yangtze River Delta Urban Agglomeration. Sustainability 2024, 16, 6624. https://doi.org/10.3390/su16156624

[15] Li, Y.; Li, C.; Feng, D. Study on Transportation Green Efficiency and Spatial Correlation in the Yangtze River Economic Belt. Sustainability 2024, 16, 3686. https://doi.org/10.3390/su16093686

[16] Su, J.; Ma, Z.; Wang, Y.; Wang, X. Evaluation and Spatial Correlation Analysis of Green Economic Growth Efficiency in Yangtze River Delta Urban Agglomeration. Sustainability 2023, 15, 2583. https://doi.org/10.3390/su15032583

[17] Lu, Y.; He, M.-e.; Liu, C. Tourism Competitiveness Evaluation Model of Urban Historical and Cultural Districts Based on Multi-Source Data and the AHP Method: A Case Study in Suzhou Ancient City. Sustainability 2023, 15, 16652. https://doi.org/10.3390/su152416652

[18] Kong, L.; Wu, Q.; Deng, J.; Bai, L.; Chen, Z.; Du, Z.; Luo, M. Assessing Regional Development Balance Based on Zipf's Law: The Case of Chinese Urban Agglomerations. ISPRS Int. J. Geo-Inf. 2023, 12, 472. https://doi.org/10.3390/ijgi12120472

[19] Jia-Bao Liu, Xu Wang, Hua Liang, Jinde Cao & Liping Chen, The Coherence and Robustness Analysis for a Family of Unbalanced Networks, IEEE Transactions on Signal and Information Processing over Networks, vol. 11, pp. 378-387, 2025. DOI. 10.1109/TSIPN.2025.3555164.

[20] Guo Y L, Hsu F C. Branding Creative Cities of Gastronomy: the role of brand experience and the influence of tourists' self-congruity and self-expansion [J]. British Food Journal, 2023, 125(8): 2803-2824.

[21] Ahn, Y.-j.; Bessiere, J. The Relationships between Tourism Destination Competitiveness, Empowerment, and Supportive Actions for Tourism. Sustainability 2023, 15, 626. https://doi.org/10.3390/su15010626

[22] Song J, Xu B. Evaluation model of urban tourism competitiveness in the context of sustainable development. Front Public Health. 2024 Jun 12;12:1396134. doi: 10.3389/fpubh.2024.1396134. PMID: 38932779

[23] Kálmán, B.G.; Grotte, J.K. The Impact of Travel and Tourism Sustainability on a Country's Image and as the Most Important Factor in the Global Competitive Index: Building Brands Based on Fogel, Schultz, and Schumpeter. Sustainability 2023, 15, 15797. https://doi.org/10.3390/su152215797

[24] Jia-Bao Liu, Lei Guan, Jinde Cao and Liping Chen, Coherence Analysis for a Class of Polygon Networks with the Noise Disturbance, IEEE Transactions on Systems Man and Cybernetics: Systems, 2025. DOI. 10.1109/TSMC.2025.3559326.

[25] Ke Y, Yang M, Xie Y. An empirical research based on spatial-temporal evolution of high-quality tourism development in Fujian Province of China. PLoS One. 2024 Dec 13;19(12):e0315221. doi: 10.1371/journal.pone.0315221. PMID: 39671463

[26] Penpece Demirer D, Büyükeke A. Unravelling tourism destination's competitiveness using big data analytics: a comparative analysis [J]. Kybernetes, 2024.

[27] Chiwaridzo O T, Masengu R. Rebuilding sustainable green tourism supply chain through technology adoption and social media branding in Zimbabwe post-COVID-19 [J]. Environment, Development and Sustainability, 2024: 1-24.

[28] Wen Y, Li Y, Zhang Y, et al. Comprehensive evaluation of global health cities development levels [J]. Frontiers in Public Health, 2024, 12: 1437647.

[29] Song J, Xu B. Evaluation model of urban tourism competitiveness in the context of sustainable development [J]. Frontiers in Public Health, 2024, 12: 1396134.

[30] Zha Q, Liu Z, Song Z, et al. A study on dynamic evolution, regional differences and convergence of high-quality economic development in urban agglomerations: A case study of three major urban agglomerations in the Yangtze river economic belt [J]. Frontiers in Environmental Science, 2022, 10: 1012304.

[31] Zhang, B.; Shao, D.; Zhang, Z. Spatio-Temporal Evolution Dynamic, Effect and Governance Policy of Construction Land Use in Urban Agglomeration: Case Study of Yangtze River Delta, China. Sustainability 2022, 14, 6204. https://doi.org/10.3390/su14106204

[32] Zhang Y, Haseeb M, Hossain M E, et al. Study on the coupling and coordination degree between urban tourism development and habitat environment in the Yangtze River Delta in China [J]. Environmental Science and Pollution Research, 2023, 30(6): 14805-14820.

[33] Tian, S.; Wang, Y.; Li, X.; Wu, W.; Yang, J.; Cong, X.; Wang, H. Spatio-Temporal Evolution and Driving Mechanism of Coupling Coordination of Pseudo Human Settlements in Central China's Urban Agglomerations. Land 2024, 13, 858. https://doi.org/10.3390/land13060858

[34] Li W, Zhang Y, Yang C, et al. Does producer services agglomeration improve urban green development performance of the Yangtze River Economic Belt in China? [J]. Ecological Indicators, 2022, 145: 109581.

[35] Chenhong X, Guofang Z. The spatiotemporal evolution pattern of urban resilience in the Yangtze River Delta city cluster based on TOPSIS-PSO-ELM [J]. Sustainable Cities and Society, 2022, 87: 104223.

[36] Pan, H.; Yang, Y.; Zhang, W.; Xu, M. Research on Coupling Coordination of China's Urban Resilience and Tourism Economy—Taking Yangtze River Delta City Cluster as an Example. Sustainability 2024, 16, 1247. https://doi.org/10.3390/su16031247

[37] Wang W, Wang S Q, Li Y S, et al. Assessing the sustainability and competitiveness of tourism economies in China's Chengdu-Chongqing metropolitan area [J]. Environmental Science and Pollution Research, 2022, 29(44): 66960-66978.

[38] Wu Y, Jia Z, Yu T. Tourism and green development: analysis of linear and non-linear effects [J]. International Journal of Environmental Research and Public Health, 2022, 19(23): 15907.

[39] Gabor M R, Kardos M, Cristache N, et al. DYNAMIC ANALYSIS OF TOURISM COMPETITIVENESS OF THE EUROPEAN COUNTRIES BASED ON DISCRIMINANT STATISTICAL ANALYSIS [J]. Economic Computation & Economic Cybernetics Studies & Research, 2021, 55(3).

[40] Díaz-Padilla, V.T.; Travar, I.; Acosta-Rubio, Z.; Parra-López, E. Tourism Competitiveness versus Sustainability: Impact on the World Economic Forum Model Using the Rasch Methodology. Sustainability 2023, 15, 13700. https://doi.org/10.3390/su151813700

[41] Tang Z, Si X, Liang Y. Research on the measurement of high-quality development of tourism: a case study of Heilongjiang Province, China [J]. Environment, Development and Sustainability, 2024, 26(10): 25027-25047.

[42] Li, S.; Cheng, Z.; Tong, Y.; He, B. The Interaction Mechanism of Tourism Carbon Emission Efficiency and Tourism Economy High-Quality Development in the Yellow River Basin. Energies 2022, 15, 6975. https://doi.org/10.3390/en15196975

[43] Zhao, X.; Mei, X.; Xiao, Z. Impact of the Digital Economy in the High-Quality Development of Tourism—An Empirical Study of Xinjiang in China. Sustainability 2022, 14, 12972. https://doi.org/10.3390/su142012972

[44] Rus, M.-I.; Munteanu, I.; Vaidianu, N.; Aivaz, K.-A. Research Trends Concerning the Danube Delta: A Specific Social-Ecological System Facing Climate Uncertainty. Earth 2025, 6, 7. https://doi.org/10.3390/earth6010007

[45] Dai X, Jiang Y, Li Y Y, et al. Evaluation of community basic public health service effect in a city in Inner Mongolia Autonomous Region——based on entropy weight TOPSIS method and RSR fuzzy set [J]. Archives of Public Health, 2023, 81(1): 149.

[46] Zuo Y, Chen H, Pan J, et al. Spatial distribution pattern and influencing factors of sports tourism resources in China [J]. ISPRS International Journal of Geo-Information, 2021, 10(7): 428.

[47] Shen Z, Zhao Q, Fang Q. Analysis of green traffic development in Zhoushan based on entropy weight TOPSIS [J]. Sustainability, 2021, 13(14): 8109.

[48] Luo J, Chen S, Sun X, et al. Analysis of city centrality based on entropy weight TOPSIS and population mobility: A case study of cities in the Yangtze River Economic Belt [J]. Journal of Geographical Sciences, 2020, 30: 515-534.

[49] Paulino I, Prats L, Domènech A. Breaking brands: New boundaries in rural destinations [J]. Sustainability, 2021, 13(17): 9921.

[50] Bian, Zhi-gang. "Environmental analysis of leisure tourism marketing situation based on cognitive environmental science." (2021): 3704-3710..

[51] Xu Q, Cheng X, Zhao H. Does the selection of high-quality scenic spots promote the growth of tourism economy evidence from China's 5A-rated tourist attractions. PLoS One. 2024 Jun 10;19(6):e0304108. doi: 10.1371/journal.pone.0304108. PMID: 38857294.

Edge-Integrated IoT and Computer Vision Framework for Real-Time Urban Flood Monitoring and Prediction

Rupesh Mandal[1], Bobby Sharma[2], Dibyajyoti Chutia[3]
Department of Computer Science and Engineering-School of Technology,
Assam Don Bosco University, Guwahati, Assam, India[1, 2]
North Eastern Space Applications Centre-Department of Space, Government of India, Umiam, Meghalaya, India[3]

***Abstract*—Urban flash floods pose a critical threat to rapidly growing cities in India, where unplanned development, climate variability, and inadequate drainage amplify risks. Guwahati, in Northeast India, experiences recurrent inundation during monsoons, disrupting livelihoods and damaging infrastructure. This study presents an integrated IoT and AI-enabled framework for urban flood monitoring and prediction. A LoRa-based IoT sensor network was deployed to capture localized hydrological and meteorological parameters, overcoming the limitations of coarse weather APIs. Rainfall forecasting was implemented at the edge layer using Random Forest, XGBoost, CatBoost, and K-Nearest Neighbors, fused through a fuzzy logic model that achieved 92.4% accuracy, surpassing individual classifiers. In parallel, a computer vision pipeline detected drainage blockages from geotagged user images, with EfficientNetB0-U-Net achieving ~91% accuracy, outperforming ResNet50, InceptionV3, and MobileNetV2. By combining rainfall prediction, IoT sensing, and blockage detection, the proposed framework delivers a holistic, low-cost, and scalable early warning system, marking a novel contribution toward resilient urban flood management in resource-constrained settings.**

***Keywords*—Urban flood prediction; IoT sensor networks; edge computing; fuzzy logic fusion; hydraulic blockage detection; computer vision**

I. Introduction

Natural hazards such as floods, tsunamis, earthquakes, and storms continue to pose significant threats worldwide, affecting human lives, resources, and infrastructure [1]. In the absence of effective monitoring and mitigation strategies, these hazards can escalate into full-scale disasters that disrupt urban life, damage infrastructure, and impede economic growth [2], [3]. Among natural hazards, floods are the most recurrent, accounting for nearly 41% of all reported disasters in the past decade [4]. Flash floods, in particular, have become increasingly destructive in urban settings. In India, such events now occur throughout the year, driven by excessive rainfall, rising water levels, and atmospheric dynamics that lead to intense precipitation [5].

Urban centers, while functioning as economic hubs, face heightened flood vulnerability due to rapid population growth, unsustainable land use, and inadequate waste and sewage management. For instance, Guwahati in Assam—located along the Brahmaputra and its tributary Bharalu—regularly experiences severe inundation in low-lying areas such as Anil Nagar and Zoo Road during the monsoon season, disrupting livelihoods and causing economic losses [6]. Land reclamation near water bodies has further aggravated seasonal flooding challenges, delaying urban development for extended periods. Although cities like Mumbai have relatively advanced flood warning and evacuation systems [7], most Indian cities, including Delhi, Bangalore, and Kolkata, lack such mechanisms.

Recent advances in the Internet of Things (IoT) have demonstrated potential for disaster monitoring and management. By integrating sensors, communication systems, and data analytics, IoT can enable real-time monitoring of hydrological parameters. However, large-scale deployment of IoT-based flood management systems remains constrained by high costs, integration complexities, limited adaptability to diverse flood scenarios, and challenges in decision-making. Despite these issues, IoT has already shown value in healthcare, defense, and disaster response. When coupled with artificial intelligence (AI), data collected from mobile devices, environmental sensors, and satellites can be transformed into actionable insights for flood prediction and mitigation.

A critical but often underexplored factor in urban flash flooding is drainage and culvert blockage. Even in well-designed drainage systems, the accumulation of debris such as boulders, vegetation, and solid waste can drastically reduce flow capacity, leading to localized inundation. Brooks [12], through scaled laboratory models and field observations, demonstrated that culvert inlets are the primary deposition points for boulders and emphasized the need for structural modifications to mitigate blockage. Building on this, Iqbal et al. [13] investigated blockage mechanisms under varying debris types, orientations, and flow conditions, highlighting how hydraulic blockage intensifies during dynamic flood hydrographs. More recently, Iqbal et al. [14] introduced a computer vision-based approach to automate culvert blockage detection, testing multiple CNN models on real-world datasets. While NASNet achieved the highest accuracy (85%) and MobileNet demonstrated faster inference, limitations such as background clutter and simplified labelling criteria constrained performance. These studies underscore the importance of integrating blockage detection into urban flood management frameworks. Manual inspection remains labor-intensive and reactive, whereas AI-driven image-based classification provides a scalable solution for real-time monitoring.

The North-Eastern region of India is particularly vulnerable, with Guwahati witnessing repeated instances of urban flooding in recent years [8]. While complete prevention of floods may not be feasible, adopting advanced technologies such as IoT and AI can significantly reduce their adverse impacts [9], [10]. IoT networks equipped with real-time sensing and analytics can provide both visual and sensor-based data, thereby enhancing early detection and timely warnings [4], [11].

The remainder of this paper is structured as follows. Section II presents a review of existing literature on IoT-based flood monitoring, machine learning for rainfall prediction, and computer vision approaches for blockage detection. Section III details the proposed methodology, including the study area, overall system architecture, IoT framework design, rainfall prediction framework, and blockage detection pipeline. Section IV discusses the results and experimental evaluations, highlighting both quantitative metrics and qualitative analyses. Section V concludes the paper with key findings and outlines potential directions for future research.

Unlike most existing studies that focus either on rainfall prediction or hydrological monitoring in isolation, this work proposes a holistic urban flood management framework that integrates three critical components: 1) a LoRa-based IoT network for real-time sensing of water levels and meteorological parameters, 2) an edge-deployed rainfall prediction system using machine learning with fuzzy fusion for improved accuracy, and 3) a computer vision–based deep learning approach for detecting blockages in drainage systems. To the best of our knowledge, this is the first research effort tailored to Guwahati city, which combines IoT and AI at both the sensing and infrastructure levels to provide a scalable, low-cost, and real-time flash flood early-warning system.

II. Literature Review

Floods are among the most destructive natural disasters worldwide, responsible for significant economic losses, ecological disruption, and fatalities [18], [19]. In India, their frequency has risen steadily due to climate change, unplanned urbanization, and population pressure [15]–[17]. According to the Global Climate Risk Index 2021, India ranks seventh among the most disaster-prone nations [16]. Between 1990 and 2020, floods accounted for more than half of the country's climate-related disasters [16], affecting millions of people each year.

Urban flooding, unlike traditional riverine floods, results primarily from excessive surface runoff in densely built environments where impermeable surfaces and inadequate drainage limit water absorption [20], [21]. This phenomenon is often described as a "hidden challenge" due to limited data availability and its localized, rapid-onset nature [4]. Unregulated settlement growth, inefficient waste management, and encroachment on natural drainage channels have intensified the vulnerability of Indian cities to flash floods [22]–[24]. Short-lived but high-intensity convective storms can overwhelm urban drainage capacity, leading to sudden waterlogging without prior warning [25]–[30]. Major Indian cities such as Mumbai, Delhi, Chennai, Hyderabad, and Surat have witnessed repeated urban flood events in recent decades, highlighting the growing severity of the problem [7].

The Northeastern Region (NER) is particularly vulnerable, with Guwahati city serving as a stark example. Positioned on the banks of the Brahmaputra River and its tributary, the Bharalu, Guwahati experiences frequent inundation in low-lying localities such as Anil Nagar and Zoo Road. These events are aggravated by rapid urban sprawl, poor drainage maintenance, and intense monsoonal rainfall [31], [8]. Reports describe floodwaters submerging key government establishments such as the State Assembly Secretariat, with some neighborhoods reporting waist-to-neck-deep water levels [8], [32]. Such recurring floods not only cause economic and infrastructural damage but also disrupt essential services and daily life.

Although complete prevention of urban floods is infeasible, their impacts can be mitigated through early-warning systems, efficient drainage monitoring, and integration of modern technologies [9]. Against this backdrop, researchers have explored IoT-based sensing, machine learning, and computer vision as emerging tools for real-time flood detection, prediction, and mitigation. The following sections review existing literature across three domains most relevant to this study: IoT-based flood monitoring frameworks, machine learning for rainfall and flood prediction, and computer vision for blockage detection in urban drainage systems.

A. IoT-Based Approaches for Urban Flood Monitoring

The Internet of Things (IoT) has been recognized as a transformative technology for environmental monitoring and disaster preparedness. By deploying water-level, rainfall, humidity, and flow sensors, IoT frameworks enable continuous, high-resolution data acquisition [4], [10], [11], [33]-[35]. LoRa/LoRaWAN has been widely adopted for its long-range, low-power communication capabilities, making it suitable for flood-prone or low-connectivity areas [36], [37]. Several studies have demonstrated IoT-based systems for flood monitoring and risk mitigation: Arshad et al. [38] emphasized the importance of IoT sensors for evacuation planning; Yang and Chang [39] integrated IoT and ML for regional inundation prediction; Li et al. [40] applied IoT with GIS for subway flood monitoring; Vitry et al. [41] introduced FloodX, combining alternative sensors and computer vision; and Soh et al. [42] developed an IoT-cloud system using image processing for severity assessment. More recently, Samikwa et al. [43] designed lightweight AI algorithms deployable on edge devices like Raspberry Pi.

Complementary to these, Bakhsh et al. [83] developed a flood forecasting model that integrates Wireless Sensor Networks (WSN), GIS, and Artificial Neural Networks (ANN). Their approach demonstrated that WSN-based solutions are not only cost-effective but also suitable for developing countries, though accuracy improvements remain necessary. Extending the scope from monitoring to prevention, Muniandy et al. [89] proposed an IoT-enabled flood prevention system capable of autonomously regulating river water flow and quality through real-time sensor networks. Their framework emphasizes preventive strategies, aligning with the Sustainable Development Goals (SDGs) on water management and climate resilience.

B. Machine Learning for Rainfall and Flood Prediction

Climate change has intensified rainfall variability, significantly increasing the frequency and severity of flash floods worldwide [44]–[58]. This challenge is particularly acute in South Asia, where the Northeast region of India experiences highly erratic precipitation patterns driven by its complex topography and monsoon dynamics. States such as Assam and Meghalaya are especially vulnerable, receiving some of the heaviest rainfall globally and experiencing recurrent flood events that disrupt infrastructure, livelihoods, and ecosystems [50]–[56]. Traditional forecasting approaches based on statistical correlations or hydrological models have made important contributions but often fail to capture the nonlinear, localized, and rapidly evolving dynamics of flash floods. Their reliance on historical datasets with coarse spatial and temporal resolution limits their ability to predict sudden high-intensity rainfall bursts, which are increasingly common under climate change.

Machine learning (ML) has emerged as a promising alternative because of its ability to model complex nonlinear relationships between environmental variables. Early applications in flood prediction applied supervised models such as Random Forests (RF), Support Vector Machines (SVM), and Artificial Neural Networks (ANN), which demonstrated stronger predictive accuracy compared to conventional regression and statistical models [61]–[64]. Ensemble methods have further improved robustness, enabling the integration of multiple weak learners to achieve higher accuracy across diverse geographies. Yang et al. [59] employed ensemble models for summer rainfall prediction in China, while Basha et al. [60] conducted a comparative analysis of deep learning and conventional ML, showing that deep models such as Long Short-Term Memory (LSTM) networks consistently outperformed classical algorithms. Similarly, Anochi et al. [65] applied ML techniques to South America, and Zhang and Ye [66] benchmarked 21 algorithms globally, both studies confirming ML's superiority over general circulation and climatological models in precipitation forecasting.

Recent works have expanded this field by exploring hybrid architectures and intelligent decision-support systems. Ali et al. [86] applied deep learning models—LSTM, Bi-LSTM, and Deep Recurrent Neural Networks (DRNN)—to predict floods in Malaysia. Their results showed that DRNN, enhanced with layer normalization and Leaky ReLU activation, provided the highest predictive accuracy and required less training time compared to LSTM and Bi-LSTM, making it more suitable for real-time applications. Complementing this purely data-driven approach, Marouane [80] introduced a distributed flood early warning framework using Multi-Agent Systems (MAS) and the Anytime Algorithm. By integrating GIS, hydrodynamic, and remote sensing data, this system reduced forecasting response times nearly fivefold, demonstrating the importance of intelligent architectures for real-time disaster management. Similarly, Md Rashid et al. [82] extended MAS-based frameworks by combining them with Case-Based Reasoning (CBR) to develop a Flood Early Warning and Response System (FEWRS). This framework not only improved technical accuracy but also explicitly considered socio-economic, environmental, and governance aspects of flood management, highlighting the need for community-centered resilience strategies.

Taken together, these studies show a clear evolution from statistical and rule-based approaches toward deep learning, hybrid, and agent-based frameworks for rainfall and flood prediction. However, key limitations remain: most ML models still rely on historical or localized datasets that do not fully capture sudden extreme events, computationally intensive deep learning approaches often hinder real-time deployment in resource-limited settings, and socio-technical integration into governance structures is still in its infancy. These gaps point toward the necessity of developing lightweight, real-time AI–IoT systems that combine efficient ML models with sensor-based data acquisition, ultimately ensuring scalable, accurate, and actionable flood prediction for vulnerable regions.

TABLE I. IOT AND ML-BASED APPROACHES FOR FLOOD/RAINFALL PREDICTION (SELECTED WORKS)

Reference #	Methods Used	IoT Device	Accuracy/Results	Key Insights / Limitations
[78]	Neuro-Fuzzy + LSTM + GA	DPS310, DHT22, Rain sensor	92.9% (Hybrid)	Advanced hybrid ML; no field IoT validation.
[77]	BiLSTM, LSTM, ARIMA	Weather sensors	BiLSTM: 92%	Accurate but resource-intensive.
[69]	Logistic Regression	DHT-11	92%	Lightweight, but scalability limited.
[68]	DT, RF, Naïve Bayes	DHT-11, BMP-180	DT: 95%, NB: 83%	Focused accuracy; lacks real-world validation.
[67]	LR, SVM, RF, DT	DHT-11, BMP-180	Up to 95% (DT)	Small-scale IoT testbed.
[71]	LSTM + PSO	NA	94%	Deep learning, computationally heavy.
[70]	MLR, Logistic Regression	Arduino + LM35	~81%	Low-cost but limited accuracy.
[74]	SVM, KNN, ANN, DNN	GPS-enabled IoT devices	ANN: 89%	Integrated sensors; energy concerns.
[73]	KNN, DT, RF, LR	IoT weather station	RF error: 0.083	IoT–ML integration; localized only.
[72]	ANN variants (FFNN, RNN, ENN)	NA	MAE: 0.54 (ENN best)	Neural nets effective; no IoT link.

The works presented in Table I demonstrate the steady evolution of IoT and machine learning applications for flood and rainfall prediction. Early studies primarily relied on simple statistical and regression models [75], but more recent works have shifted towards neural networks, deep learning, and hybrid architectures [77], [78], [76]. Distributed and agent-based frameworks [80], [92] have further advanced the field by improving decision-making and integrating governance

perspectives, while IoT-driven monitoring and prevention solutions [99], [112] highlight the move toward practical, cost-effective deployment. Despite these advances, most frameworks remain constrained to localized datasets, simulation-based validation, or resource-intensive models, underscoring the need for lightweight, field-deployable AI-IoT systems that integrate hydrological sensing with infrastructure monitoring.

C. Computer Vision for Blockage Detection in Urban Drainage Systems

Blockages in culverts and drainage channels are a critical yet underexplored factor in urban flooding, as even well-designed drainage systems can fail when debris accumulation reduces their flow capacity. Unlike hydrological or rainfall-based causes of flooding, blockage issues often arise from localized debris such as vegetation, sediments, plastics, and boulders, which can obstruct culvert inlets or narrow drainage passages. Early investigations by Balkham et al. [79] addressed this problem through a risk-based framework in the United Kingdom, providing local-scale guidelines for blockage management, while Kramer et al. [81] extended this line of work by introducing mathematical models and laboratory-scale experiments to quantify the effects of different debris types and alignments. Although valuable, these approaches were often limited by oversimplification, with restricted applicability to real-world conditions. Field-supported laboratory evidence by Brooks [12] later confirmed that culvert inlets are the primary points of debris deposition, particularly boulders, thereby validating the vulnerability of structural entry points during intense flows.

With the advent of advanced computing, researchers shifted toward AI-driven solutions for blockage assessment. Iqbal et al. [13] proposed computer vision methods using convolutional neural networks (CNNs) to automate blockage classification, testing architectures such as NASNet, MobileNet, and others on real-world and laboratory datasets. Their results demonstrated that NASNet achieved the highest accuracy of around 85%, while MobileNet provided faster inference speeds, making it suitable for near-real-time applications. In a follow-up study, Iqbal et al. [14] extended this work by integrating CNN-extracted features into regression models such as Artificial Neural Networks (ANNs) to predict hydraulic blockage, thereby attempting to link visual blockage data with quantitative hydrological behavior. This represented a significant step forward, as it bridged the gap between visual inspection and hydraulic modeling.

Despite these advances, challenges remain. Computer vision models often struggle with background clutter, varying illumination, occlusion of culvert openings, and the scarcity of high-quality, annotated datasets. Furthermore, most existing approaches have been validated only in laboratory conditions or using limited synthetic datasets, raising concerns about their robustness in complex real-world environments. Another critical gap is the lack of integration between image-based blockage detection systems and IoT sensor networks, which could enable holistic urban flood monitoring by combining hydraulic, meteorological, and infrastructural indicators. Addressing these challenges through the development of scalable, data-rich, and integrated AI–IoT frameworks is essential for achieving reliable and proactive blockage detection in urban flood management.

As shown in Table II, research on blockage detection in culverts and drainage systems has advanced from conceptual risk frameworks [79], [81] to AI-driven visual recognition methods [13], [14]. More recent contributions employ deep learning pipelines, including segmentation-classification approaches and semantic segmentation architectures such as SHARP-Net. Despite notable improvements in accuracy, most of these studies remain limited to controlled laboratory experiments or narrowly scoped datasets (e.g. ICOB, VHD, S-BIRD). Real-world deployment challenges—including background clutter, occlusions, lighting variability, and lack of integration with IoT sensor networks—remain largely unresolved. These limitations highlight a critical research gap: the need for a scalable and integrated system that combines IoT-based sensing with image-based blockage detection for comprehensive urban flood monitoring.

TABLE II. COMPUTER VISION AND ML APPROACHES FOR BLOCKAGE DETECTION (SELECTED WORKS)

Reference #	Methods Used	Dataset	Accuracy/Results	Key Insights / Limitations
[86]	SHARP-Net (semantic segmentation)	Culvert-Sewer, DeepGlobe	IoU: 94.7%	Struggles under occlusion.
[87]	Siamese NN + Binary classifier	80k UK trash screen images	Acc: 91%, AUC: 0.98	Needs hydro model integration.
[88]	Logistic Regression	Trash screen CCTV	88%	Dataset bias (80% blocked).
[85]	YOLOv5	S-BIRD sewer dataset	mAP 96.3%	Sewer-focused, not culverts.
[84]	KNN, ANN, SVR, 1D-CNN	HBD dataset	ANN: $R^2 = 0.95$	Strong prediction; lab-scale only.
[14]	ANN + MobileNet features	Lab culvert dataset	$R^2 = 0.7855$	Weak link visual ↔ hydraulic.
[13]	CNNs (NASNet, MobileNet, etc.)	ICOB, VHD, SIC	NASNet: 85%, MobileNet: 78%	Clutter & labeling issues.
[12]	Lab + field blockage study	Culvert inlet models	Inlets = primary deposition	Focused only on boulders.
[81]	Mathematical model + experiments	Lab culvert debris tests	Showed debris type/alignment effects	Simplified; lacked real-world validation.
[79]	Risk-based blockage framework	Field data (UK culverts)	Guidelines for blockage mgmt.	Early risk framework; no automation.

III. Proposed Methodology

A. Study Area

The study was conducted in Anil Nagar and Zoo Road localities of Guwahati City, Assam, India, one of the most flood-prone urban centers in Northeast India. Guwahati lies along the Bharalu River, a tributary of the Brahmaputra, and experiences a humid subtropical climate with intense monsoonal rainfall (June–September). Anil Nagar, a densely populated residential and commercial zone with low-lying terrain, frequently suffers from flash floods due to inadequate drainage and rapid urbanization. Heavy rainfall causes sharp rises in water levels, leading to recurrent inundation, waterlogging, and economic losses. For this study, a pilot area in Kamrup Metropolitan District was selected to test the proposed IoT- and vision-based flood monitoring system. The selected sites are shown in Fig. 1.

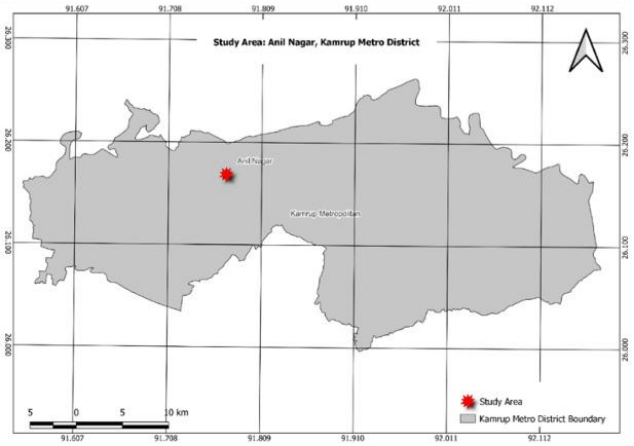


Fig. 1. Study area map: Anil Nagar in Kamrup Metro District.

B. Overall Architecture

The overall process flow of the proposed methodology is illustrated in Fig. 2. The framework integrates IoT-based sensing, machine learning-driven rainfall prediction, and computer vision-based blockage detection into a unified flood monitoring and early warning system.

A LoRa-based IoT network was deployed across the study area using two types of devices—water sensing nodes and base stations. Water sensing nodes monitor drainage water levels, while base stations collect meteorological parameters such as rainfall, temperature, and air pressure. These devices transmit real-time data wirelessly for further processing.

The acquired IoT data, along with historical datasets from external APIs, are pre-processed and analyzed using multiple machine learning algorithms for rainfall prediction. Their outputs are refined through a fuzzy logic-based fusion framework, which demonstrated superior accuracy compared to individual classifiers. By executing these tasks at the edge, the system achieves low latency, improved reliability, and reduced dependence on cloud-only computation.

At the cloud layer, images of strategic canal locations where blockages are most likely to occur are captured and uploaded by users through the mobile application. The system incorporates geofencing, which triggers an alert when users enter blockage-prone zones, prompting them to capture and submit images. Importantly, this process is designed to occur before flood events so that blockages can be identified proactively when rainfall is predicted, thereby minimizing reliance on user activity during emergencies. Once uploaded, these images are processed on the server using computer vision-based deep learning models to detect and classify blockages. By integrating blockage detection results with rainfall forecasts and IoT-based monitoring data, the cloud layer enables a comprehensive flash flood prediction and early warning system for the selected localities.

Finally, the mobile application and web dashboard provide stakeholders with real-time visualizations, alerts, and decision-support tools. The app displays live water levels, rainfall forecasts, and blockage alerts, while also offering geofencing-based warnings and safe-route suggestions. The dashboard aggregates all information, ensuring that authorities and citizens have timely access to actionable flood intelligence.

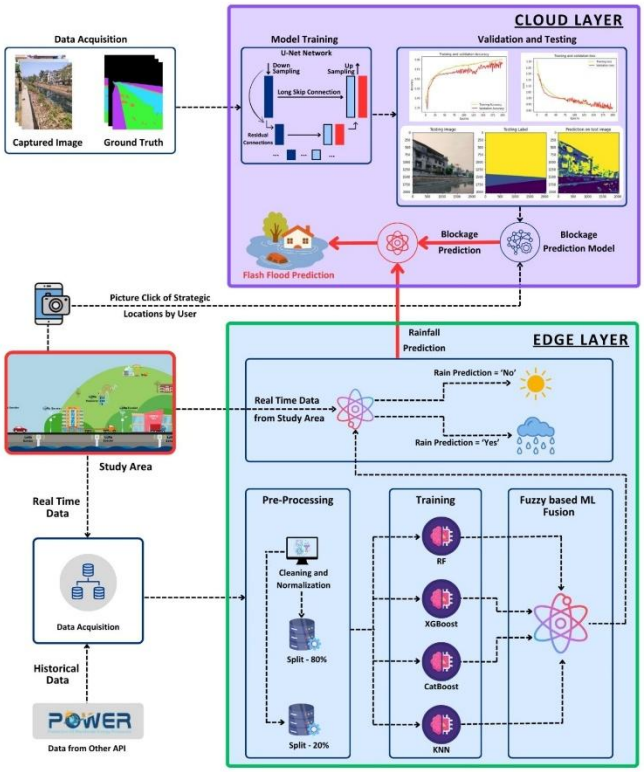


Fig. 2. Overall architecture of the proposed work.

C. IoT Framework and Hardware Design

The IoT framework was designed to provide real-time monitoring of drainage water levels and meteorological parameters within the study area. The system comprises two main devices: water sensing nodes and base stations.

The water sensing nodes are compact instruments deployed along roadside drains and low-lying urban locations to detect potential flash flood events. Each node is equipped with a contact-type water level sensor, enclosed within PVC pipes positioned 200 cm apart for accurate depth measurements. To enable reliable communication in flood-prone environments, the nodes employ LoRa transceivers, which facilitate real-time data

transmission across long ranges. Unlike conventional IoT devices dependent on internet access, this system remains operational even during disasters when internet connectivity is disrupted, thereby ensuring uninterrupted flood monitoring. Designed for autonomous operation, the nodes are powered by rechargeable battery packs integrated with solar panels, eliminating dependence on conventional power supplies. This ensures continuous functionality even during prolonged power outages, often associated with flood scenarios.

The base stations, developed using Raspberry Pi 4 boards, serve as gateways for aggregating data from multiple sensing nodes. In addition to LoRa receivers, they are equipped with BMP180 pressure sensors, tipping rain gauges, anemometers, and temperature–humidity sensors, enabling on-site weather monitoring. These parameters were purposely included because relying solely on weather APIs can be misleading, as such services typically provide averaged values over large geographic areas. In reality, micro-climatic variations in parameters such as rainfall, wind speed, and humidity can occur within ranges as small as 4–5 km. For flash flood predictions, localized rainfall data is particularly critical, since heavy rainfall may occur in one neighborhood while an adjacent area just 2 km away remains unaffected. The incorporation of direct measurements at the base stations, therefore, ensures high-resolution, site-specific meteorological data, significantly improving the accuracy of the overall flood monitoring and prediction system.

This hardware architecture, integrating low-power LoRa communication with solar-powered autonomy and localized weather sensing, establishes a scalable, resilient, and energy-efficient flood monitoring system suitable for deployment in vulnerable urban environments. Fig. 3 illustrates the deployment of IoT sensors in the study area, showing the placement of water sensing nodes and base stations across critical flood-prone locations.



Fig. 3. Deployment of nodes and base stations.

D. Edge-Based Rainfall Prediction Using Machine Learning Fusion

Accurate rainfall prediction is essential for flash flood forecasting, as localized precipitation patterns often determine the severity of inundation. In this study, rainfall prediction was developed by integrating historical meteorological datasets with real-time IoT sensor data collected from the deployed framework in the study area. Ten years of historical data were obtained from the Indian Meteorological Department (IMD) and the NASA POWER database, covering rainfall, temperature, humidity, and pressure parameters. These datasets were merged with on-site measurements from IoT base stations to provide both temporal depth and spatial specificity, addressing limitations of weather APIs that only provide coarse, area-wide averages. As the IoT base stations continue to record location-specific data, the dataset will grow continuously, enabling future retraining of the models. This evolving dataset is expected to improve predictive accuracy and adaptability, including unprecedented rainfall patterns that may arise due to climate change.

The complete dataset comprised 91,991 instances with seven features (six independent variables and one dependent variable). Pre-processing involved three major steps: 1) cleaning, where missing values were handled using mean imputation; 2) normalization, which standardized feature ranges; and 3) splitting, where the dataset was divided into training and testing subsets in an 80:20 ratio. These steps enhanced the classifiers' ability to learn effectively and minimized bias.

Four machine learning algorithms were evaluated for rainfall prediction: Random Forest (RF), Extreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and K-Nearest Neighbors (KNN). These models were iteratively optimized during both training and testing to capture the nonlinear and multivariate patterns of rainfall events. To further improve reliability, the outputs from individual classifiers were passed through a fuzzy logic-based fusion framework. Rather than assigning rainfall into discrete categories, the fuzzy layer refines the combined outputs of multiple models to increase prediction probability and confidence. A decision matrix (Table III) was designed to integrate model outputs into a unified outcome.

TABLE III. INPUT INTO THE FUZZY LAYER FOR THE PROPOSED MODEL

Individual Model Prediction				Rainfall Prediction
RF	***XGBoost***	***CatBoost***	***KNN***	
Yes	Yes	Yes	Yes	**Yes**
Yes	Yes	Yes	No	**Yes**
Yes	Yes	No	Yes	**Yes**
Yes	Yes	No	No	**Yes**
Yes	No	Yes	Yes	**Yes**
Yes	No	Yes	No	**Yes**
Yes	No	No	Yes	**Yes**
Yes	No	No	No	**No**
No	Yes	Yes	Yes	**Yes**
No	Yes	Yes	No	**No**
No	Yes	No	Yes	**No**
No	Yes	No	No	**No**
No	No	Yes	Yes	**No**
No	No	Yes	No	**No**
No	No	No	Yes	**No**
No	No	No	No	**No**

A key feature of this work is that the machine learning models and the fuzzy fusion framework were deployed on Raspberry Pi devices at the edge layer. This deployment enables on-site rainfall prediction with low latency, reducing

dependence on cloud servers and ensuring functionality even in disaster scenarios where internet connectivity may be disrupted. Edge-level prediction thus allows the system to provide faster and more reliable early warnings, an essential requirement for flash flood-prone urban environments. Moreover, the architecture has been designed in a modular manner, allowing additional nodes, base stations, or processing units to be integrated seamlessly. This modularity supports future scalability to larger urban areas or more complex drainage networks without fundamentally altering the core framework, thereby making the system adaptable for broader deployment.

The experimental results demonstrated that the fuzzy fusion model consistently outperformed standalone classifiers, producing more stable and accurate rainfall forecasts while reducing false positives. As illustrated in Fig. 4, by combining IoT-based observations, historical meteorological records, and intelligent fusion of multiple classifiers at the edge, this rainfall prediction module forms a critical component of the proposed framework, directly supporting an integrated flash flood early warning system for Guwahati's vulnerable localities.

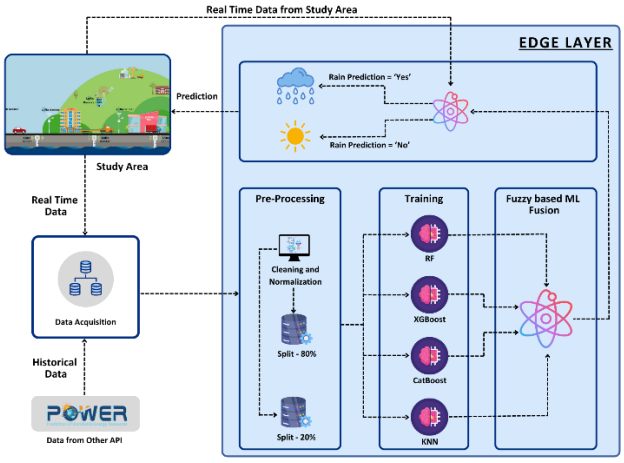


Fig. 4. Workflow of the rainfall prediction framework integrating IoT data and historical meteorological datasets. The figure illustrates data pre-processing, machine learning classification (RF, XGBoost, CatBoost, KNN), and fuzzy logic-based fusion, all deployed on Raspberry Pi devices at the edge layer to enable real-time, low-latency rainfall prediction and early warning.

E. Computer Vision–Based Blockage Detection

Blockages in urban drainage canals significantly reduce hydraulic capacity and are among the leading causes of flash flooding in Guwahati. To address this, a computer vision-based blockage detection module was developed and integrated into the proposed framework. A custom dataset comprising 1,130 high-resolution images of drainage canals and outlets was collected from strategic locations within the Anil Nagar and Zoo Road localities. Images were captured using smartphone cameras (Realme 9 Pro 5G) under varying lighting and environmental conditions to represent real-world variability. Each image, originally sized at 3468 × 4624 pixels, was resized to 1920 × 2560 pixels and then divided into non-overlapping patches of 512 × 512 pixels, yielding a total of 16,950 samples for training and evaluation. The dataset was annotated using the CVAT (Computer Vision Annotation Tool), where each patch was manually segmented into four classes (Table IV):

TABLE IV. SEGMENTATION CLASS LABLES

Sl. No.	Class	Color Labels
1	Water	#33DDFF
2	Boundary	#B83DF5
3	Blockage	#FF6A4D
4	Background	#000000

The careful assignment of these classes ensured that canal regions, potential obstruction zones, and their surroundings were distinctly represented, enabling the model to effectively learn spatial differences between clear flow areas and blocked sections. Fig. 5 shows sample annotated images, where the original canal image is placed on the left and the corresponding segmentation mask on the right, highlighting the four defined classes.

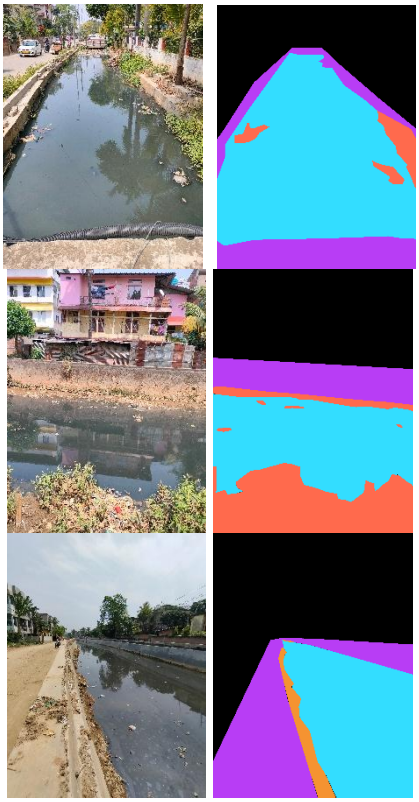


Fig. 5. Sample dataset images with annotations. Left: original canal images. Right: annotated segmentation masks showing four classes (Water, Boundary, Blockage, Background).

The original images were captured at a high resolution of 3468 × 4624 pixels. Due to computational resource limitations, they were first resized to 1920 × 2560 pixels and subsequently divided into patches of 512 × 512 pixels, producing a total of 16,950 samples. This approach preserved greater global context compared to lower-resolution alternatives, while ensuring manageable input sizes for the models. The resulting patches were normalized and augmented using random rotation, brightness adjustment, flipping, and noise injection. The dataset was then split into 70% training, 15% validation, and 15% testing subsets.

For segmentation, four deep learning models were trained and evaluated to identify blockages: a U-Net with EfficientNetB0 backbone, a U-Net with ResNet50 backbone, a U-Net with InceptionV3 backbone, and a U-Net with MobileNetV2 backbone. The U-Net framework was selected for its strong performance in semantic segmentation, while the different backbones were used to investigate trade-offs in feature representation, computational efficiency, and accuracy. All models were trained under identical conditions to ensure fair comparison.

As shown in Fig. 6, the encoder backbone incorporates MBConv blocks, where each block first expands the number of channels using 1×1 convolutions, denoted as Ex(.) in Eq. (3). This is followed by a Depthwise Convolution D(.) and refinement using a Squeeze-and-Excitation attention module Se(.). Finally, another set of 1×1 convolutions, represented as Cn(.), compresses the channel count back, forming the well-known "inverted bottleneck" structure. Residual connections are integrated to improve gradient propagation and training stability.

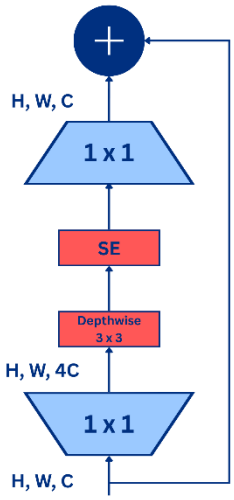


Fig. 6. Illustration of the MBConv block and skip connection mechanism used in the encoder–decoder of U-Net backbones.

The decoder mirrors this design, with upsampling operations that reconnect to the encoder via skip connections. Eq. (3) and (4) describe the feature propagation in the encoder and decoder stages. Here, X_{En}^{i} denotes the feature map produced by the i^{th} MBConv encoder block, while X_{De}^{j} and X_{En}^{j} represent the feature maps from the j^{th} decoder and encoder blocks, respectively. Skip connections are combined using the CBAR(.) function, which merges the encoder's feature map with the upsampled feature map from the preceding decoder block. The merged output is then processed through a 3×3 convolution (C3×3(.)), batch normalization (BN(.)), an activation function δ(.), and a residual connection, as expressed in Eq. (5). This design ensures better gradient flow, higher feature reuse, and robust segmentation performance.

All models were implemented in TensorFlow 2.10 and trained using an Nvidia RTX 3080 GPU (10,496 CUDA cores, 10GB GDDR6X memory). Training was performed for 100 epochs with a batch size of 16, using the Nadam optimizer (learning rate = 0.001). A hybrid loss function combining Dice Loss and Focal Loss was applied to maximize segmentation overlap and address class imbalance.

Once deployed on the cloud server, the trained models process geotagged images uploaded via the mobile application. When a user enters a geofenced high-risk drainage zone, the application prompts them to capture and submit canal images. The models then classify the canal as "blocked" or "clear". Hosting the computer vision models on the cloud ensures not only centralized processing but also inherent scalability, as elastic computing resources can accommodate larger datasets and concurrent image submissions from wider geographical areas. The comparative performance of the four segmentation approaches is presented in the Results and Discussion section.

$$X_{En}^{i} = Cn\left(Se\left(D\left(Ex\left(X_{En}^{i-1}\right)\right)\right)\right) + X_{En}^{i-1} \quad (1)$$

$$X_{De}^{j} = CBAR\left(concat\left(X_{En}^{j}, U\left(X_{De}^{j-1}\right)\right)\right) \quad (2)$$

$$CBAR(X) = \delta(BN(C_{3X3}(X)) + X \quad (3)$$

IV. Results and Discussion

A. IoT Deployment and Field Validation

The proposed IoT framework was deployed in Anil Nagar and Zoo Road localities of Guwahati, with 10 water sensing nodes and 2 base stations strategically placed along drainage channels and flood-prone areas. Each water sensing node was equipped with LoRa transceivers and solar-powered batteries, while the base stations collected supplementary meteorological parameters, including rainfall, temperature, humidity, and wind speed.

To ensure real-world practicality, the proposed framework employs solar-powered sensing nodes equipped with 7500 mAh lithium-ion battery packs that are continuously recharged during the day by integrated solar panels. This design ensures uninterrupted operation even during prolonged cloudy or rainy conditions, thereby addressing energy constraints often faced by IoT systems. Coupled with low-power LoRa communication modules, the nodes achieve efficient energy usage and reliable data transmission. The hardware requirements remain modest, with commodity sensors and Raspberry Pi-based edge nodes keeping costs affordable for larger deployments. The modular architecture further supports integration of heterogeneous devices, enabling scalability in dense urban environments. Fault tolerance and data reliability are enhanced through redundant data logging at both node and base-station levels, ensuring continuity of operations even in cases of sensor malfunction or communication disruptions.

To validate the field data, real-time sensor outputs were compared with the NASA POWER API and Indian Meteorological Department datasets. The results showed a high degree of consistency, with local rainfall measurements aligning closely with API estimates but offering finer spatial granularity. This confirmed that localized sensors captured micro-climatic variations more effectively than coarse-resolution APIs, a critical factor for flash flood prediction in Guwahati, where rainfall can vary significantly within 2–3 km.

A representative comparison between IoT-based rainfall measurements and NASA POWER API values for a sample week in July 2023 is presented in Table V. The results highlight that IoT sensors captured short-duration rainfall bursts and localized variations which were not fully reflected in API data, demonstrating the added value of in-situ measurements for urban flood forecasting.

TABLE V. VALIDATION OF SENSOR DATA FROM THE ZOO ROAD BASE STATION AGAINST NASA POWER API VALUES (10–16 JULY 2023)

Date	IoT Sensor Rainfall (mm)	NASA POWER API (mm)	Difference (mm)
10-Jul-23	54.2	52.8	1.4
11-Jul-23	12.6	11.9	0.7
12-Jul-23	0	0	0
13-Jul-23	38.7	40.1	–1.4
14-Jul-23	22.1	23.3	–1.2
15-Jul-23	5.8	6.4	–0.6
16-Jul-23	47.5	46.2	1.3

During the 2023 monsoon season, the deployed network successfully detected rapid increases in water levels along the Bharalu tributary, providing early indications of waterlogging in Anil Nagar. As illustrated in Fig. 7, the water sensing nodes captured a clear rise in drainage levels, validating the effectiveness of the deployed sensors. These results demonstrate the reliability of the IoT network for continuous monitoring, even under adverse weather conditions, and its capacity to generate high-resolution hydrological datasets essential for predictive modeling.

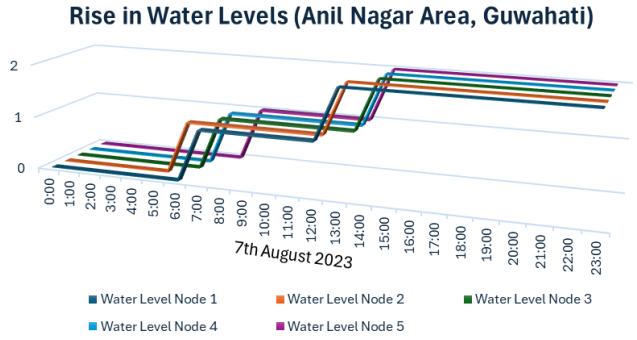
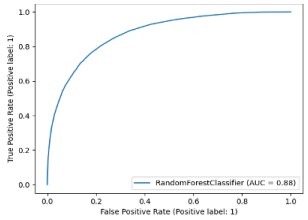


Fig. 7. Rise in water levels as depicted by the water sensing nodes

B. Rainfall Prediction Performance

The time series data of weather parameters were analyzed for the development of the rainfall prediction system. Datasets were obtained from the Indian Meteorological Department (two years) and NASA POWER (ten years) for Guwahati city. Several classification algorithms, including Random Forest (RF), XGBoost, CatBoost, and K-Nearest Neighbors (KNN), were tested to evaluate their predictive performance.

During the testing phase, the Random Forest (RF) model correctly classified 8094 out of 9261 negative instances and 8213 out of 9138 positive instances, yielding an accuracy of 88.6% with a miss rate of 11.4%. The comparison between expected and actual outcomes is shown in Fig. 8. The XGBoost model produced similar results, with 8103 negatives and 8134 positives classified correctly, corresponding to an accuracy of 88.2% and an 11.8% miss rate, as illustrated in Fig 9. In comparison, CatBoost achieved the best standalone performance, predicting 8241 negatives and 8231 positives, resulting in an accuracy of 89.5% and a 10.5% miss rate, as shown in Fig 10. By contrast, the KNN model exhibited weaker performance, with 7213 negatives and 7311 positives correctly classified, yielding 78.9% accuracy and a 21.1% miss rate, as depicted in Fig. 11.

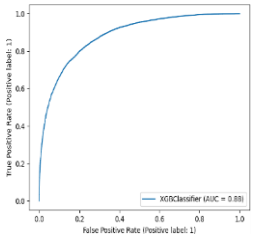


RF Testing Results

No. of samples		N= 18399	
	Expected Result (E)	Output Result (O)	
		O-Negative	O-Positive
Input	E-Negative 9261	8094	1167
	E-Positive 9138	925	8213

Accuracy=	0.886298168
Miss Rate=	0.113701832

Fig. 8. Statistical analysis for the random forest model

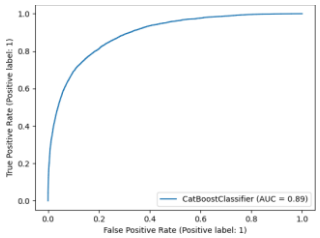


XGBoost Testing Results

No. of samples		N= 18399	
	Expected Result (E)	Output Result (O)	
		O-Negative	O-Positive
INPUT	E-Negative 9261	8103	1158
	E-Positive 9138	1004	8134

Accuracy=	0.882493614
Miss Rate=	0.117506386

Fig. 9. Statistical analysis for a random XGBoost model

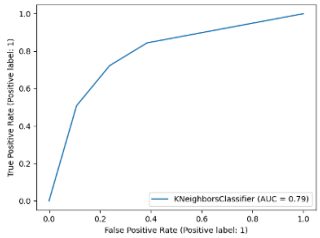


CatBoost Testing Results

No. of samples		N= 18399	
	Expected Result (E)	Output Result (O)	
		O-Negative	O-Positive
INPUT	E-Negative 9261	8241	1020
	E-Positive 9138	907	8231

Accuracy=	0.895266047
Miss Rate=	0.104733953

Fig. 10. Statistical analysis for the CatBoost model



KNN Testing Results

No. of samples		N= 18399	
	Expected Result (E)	Output Result (O)	
		O-Negative	O-Positive
INPUT	E-Negative 9261	7213	2048
	E-Positive 9138	1827	7311

Accuracy=	0.789390728
Miss Rate=	0.210609272

Fig. 11. Statistical analysis for the K-Nearest Neighbor model

Proposed Fusion model Testing Results

No. of samples		N= 18399	
	Expected Result (E)	Output Result (O)	
		O-Negative	O-Positive
INPUT	E-Negative 9261	8347	914
	E-Positive 9138	486	8652

Accuracy=	0.923908908
Miss Rate=	0.076091092

Fig. 12. Statistical analysis for the proposed fusion model

Subsequently, the outputs of all classifiers were integrated using the proposed fuzzy fusion framework. This system processed the test data by considering actual class labels alongside predictions from individual classifiers. The fuzzy model identified 8347 negatives (out of 9216) and 8652 positives (out of 9138), achieving an overall accuracy of 92.4% with a miss rate of 7.6%, as illustrated in Fig. 12. This demonstrates the advantage of the fusion approach in reducing false classifications and improving robustness compared to standalone models.

A detailed summary of training and testing results across all models, along with the fuzzy fusion framework, is presented in Table VI. Notably, the fused system consistently outperformed the four individual classifiers, confirming its effectiveness in rainfall prediction. Furthermore, Table VI also includes a comparative evaluation against previously reported rainfall prediction approaches, highlighting the superior accuracy and reduced miss rate of the proposed methodology.

TABLE VI. COMPARISON OF MACHINE LEARNING MODELS AND FUZZY FUSION FRAMEWORK FOR RAINFALL PREDICTION

Model	Accuracy (%)	Miss Rate	Remarks
Random Forest (RF)	88.6	0.14	Strong generalization, moderate error rate
XGBoost	88.2	0.12	Effective but prone to overfitting with small samples
CatBoost	89.5	0.1	Best standalone classifier
KNN	78.9	0.21	Sensitive to noise, weakest performance
Fuzzy Fusion	92.4	0.08	Highest accuracy, reduced false predictions

C. Blockage Detection via Deep Learning

The efficacy of various deep learning architectures was evaluated for the task of blockage detection in canal networks. Four models, like U-Net with EfficientNetB0 backbone, U-Net with ResNet50 backbone, U-Net with InceptionV3 backbone, and U-Net with MobileNetV2 backbone were trained and tested on a dataset of 1,130 high-resolution images (3468 × 4624 pixels) annotated into blockage and non-blockage classes. To address computational constraints and increase the effective dataset size, the images were resized to 1920 × 2560 pixels and divided into non-overlapping 512 × 512 patches, resulting in 16,950 samples. These patches were normalized and augmented to enhance robustness and variability. Model training was conducted in a high-performance GPU environment (NVIDIA RTX 3080), enabling efficient parallel computation and accelerated convergence. The performance of the models was evaluated using standard metrics, including Jaccard's coefficient (IoU), mean Intersection-over-Union (mIoU), Dice coefficient, pixel accuracy, precision, and error rate. The U-Net with EfficientNetB0 backbone achieved the best results, with a pixel-wise accuracy of 91%, Dice score of 86%, and mIoU of 78%. This superior performance can be attributed to EfficientNet's inverted bottleneck blocks and squeeze-and-excitation modules, which provided effective feature extraction while retaining spatial details through U-Net's skip connections. The combination allowed the model to segment small or irregular blockages with higher precision than other backbones.

$$\text{IoU} = \frac{TP}{TP+FP+FN} \tag{4}$$

$$\text{mean IoU} = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i+FP_i+FN_i} \tag{5}$$

$$\text{Dice} = \frac{2 \cdot TP}{2 \cdot TP+FP+FN} \tag{6}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

$$\text{Error Rate} = \frac{FP+FN}{TP+TN+FP+FN} \tag{8}$$

The terms TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively, and are fundamental in understanding the performance of classification models.

The ResNet50 backbone ranked second, with an accuracy of 85%. Its deep residual connections supported robust feature extraction and strong generalization, but the absence of symmetric skip connections in its native architecture limited precise boundary reconstruction. InceptionV3 achieved 72% accuracy, leveraging multi-scale convolutional kernels for contextual understanding but failing to produce sharp pixel-wise predictions due to the lack of a dense decoder path. At the lower end, MobileNetV2 achieved only 65% accuracy, reflecting its limited representational power as a lightweight architecture optimized for efficiency rather than dense segmentation.

Dataset characteristics also influenced performance. The modest dataset size (1130 images) and class imbalance, where blockage pixels formed only a small fraction of each image, caused models to be biased toward background predictions. Manual annotations introduced additional variability, particularly in ambiguous regions such as shadows or partially submerged debris. While EfficientNetB0-U-Net was relatively robust to such noise, MobileNetV2 and InceptionV3 were more affected. Resizing high-resolution images to 512 × 512 patches also led to loss of fine details, reducing the ability of weaker backbones to detect small debris.

TABLE VII. COMPARATIVE ANALYSIS OF EXISTING CLASSIFICATION MODELS WITH THE SEGMENTATION-BASED MODELS

Model	Pixel Accuracy	Mean IOU	Dice Coefficient	Precision	Recall
MobileNetV2	65%	40%	55%	58%	52%
InceptionV3	72%	50%	63%	65%	60%
ResNet-50	85%	68%	78%	80%	75%
EffecientNetB0	**91%**	**78%**	**86%**	**88%**	**85%**

Training and validation accuracy curves over 100 epochs (Fig. 13) show that the EfficientNetB0-U-Net achieved smooth convergence, with training accuracy reaching ~93% and validation accuracy ~90%, indicating minimal overfitting. ResNet50 also demonstrated stable learning, converging around 87% training accuracy and 84% validation accuracy. InceptionV3 showed slower convergence, plateauing at ~73% training accuracy and ~70% validation accuracy, while MobileNetV2 displayed severe overfitting, with training accuracy climbing to ~66% but validation accuracy stagnating at ~30%. These trends are further supported by the comparative results summarized in Table VII, which highlight the superior performance of EfficientNetB0-U-Net over the other backbones.

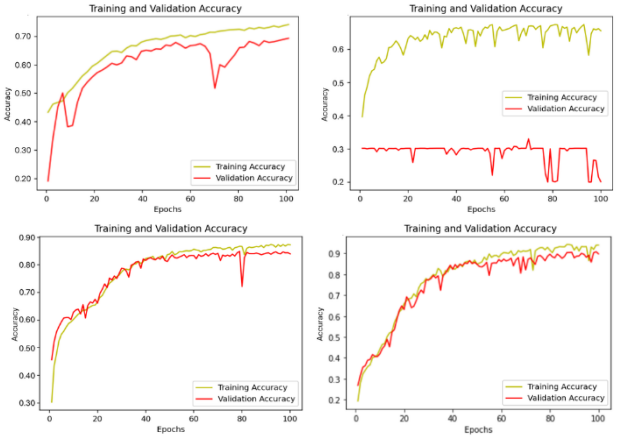


Fig. 13. Training and validation accuracy curves over 100 epochs for the four models: Top Left – MobileNetV2-U-Net; Top Right – InceptionV3-U-Net; Bottom Left – ResNet50-U-Net; Bottom Right – EfficientNetB0-U-Net

Loss curves: Fig. 14 further confirms these observations. EfficientNetB0-U-Net and ResNet50 showed steadily declining training and validation loss with minimal divergence, reflecting robust learning. InceptionV3's validation loss exhibited early instability with sharp spikes before stabilizing, while MobileNetV2 showed highly erratic validation loss throughout, indicating poor generalization.

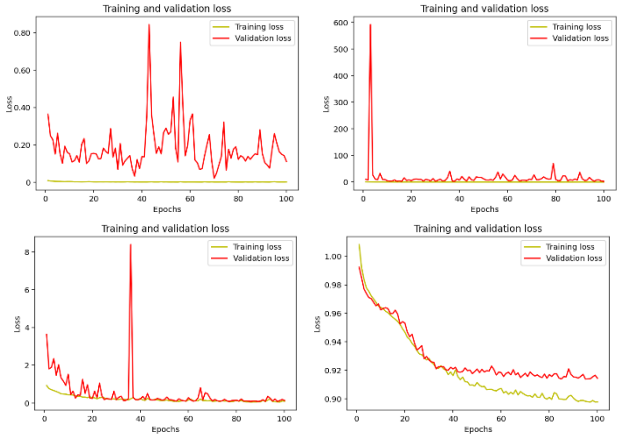


Fig. 14. Training and validation Loss curves over 100 epochs for the four models: Top Left – MobileNetV2-U-Net; Top Right – InceptionV3-U-Net; Bottom Left – ResNet50-U-Net; Bottom Right – EfficientNetB0-U-Net

Sample qualitative outputs (Fig. 15–18) further reinforce the quantitative findings. To examine model performance under different conditions, three representative test images were chosen, corresponding to low, moderate, and high blockage levels. Predictions were then generated using four models—MobileNetV2-U-Net (Fig. 15), InceptionV3-U-Net (Fig. 16), ResNet50-U-Net (Fig. 17), and EfficientNetB0-U-Net (Fig. 18)—and compared against their respective ground truth masks.

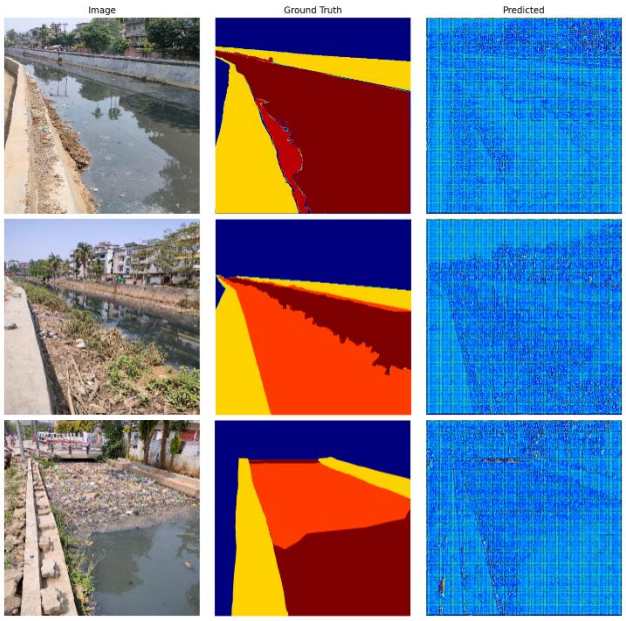


Fig. 15. Predicted segmentation outputs using MobileNetV2-U-Net. Top: low blockage; Middle: moderate blockage; Bottom: high blockage, shown alongside ground truth masks.

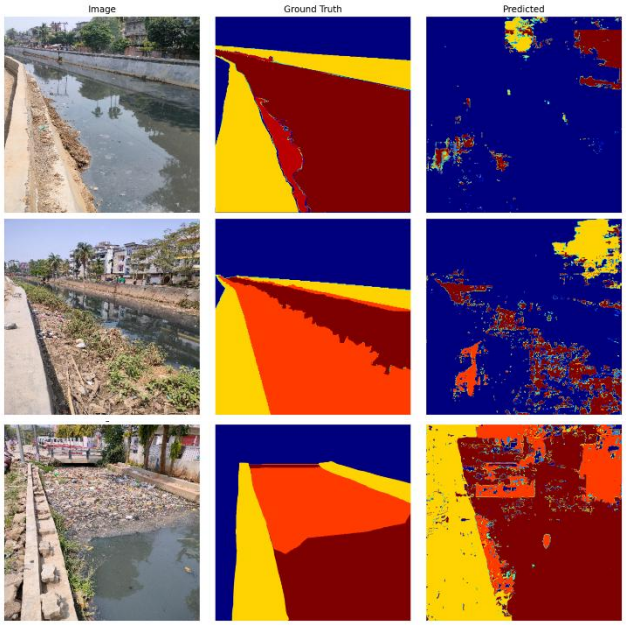


Fig. 16. Predicted segmentation outputs using InceptionV3-U-Net. Top: low blockage; Middle: moderate blockage; Bottom: high blockage, shown alongside ground truth masks.

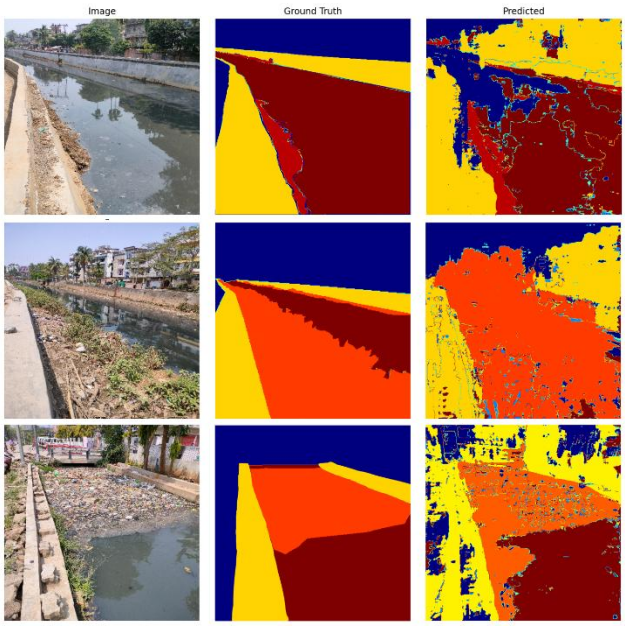


Fig. 17. Predicted segmentation outputs using ResNet50-U-Net. Top: low blockage; Middle: moderate blockage; Bottom: high blockage, shown alongside ground truth masks.

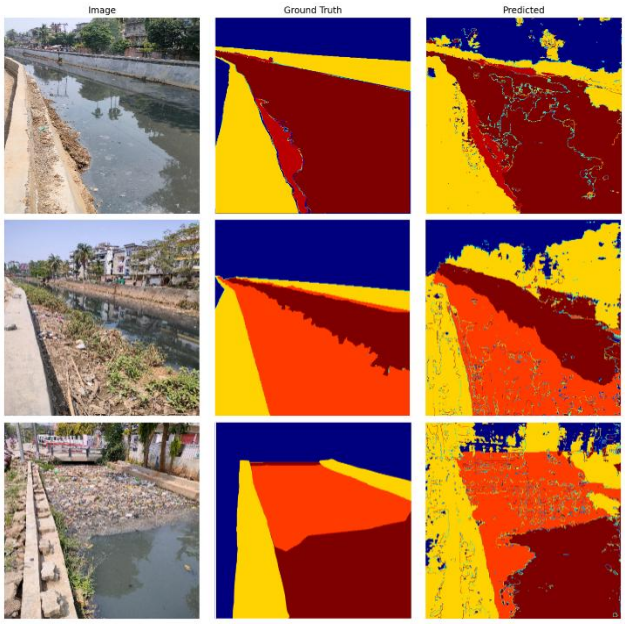


Fig. 18. Predicted segmentation outputs using EfficientNetB0-U-Net. Top: low blockage; Middle: moderate blockage; Bottom: high blockage, shown alongside ground truth masks.

Among these, EfficientNetB0-U-Net consistently produced segmentation maps that most closely resembled the ground truth. It excelled at detecting small debris, partial blockages, and irregular boundary regions, which are often the most challenging to segment in real-world canal imagery. This effectiveness can be attributed to the synergy of EfficientNet's inverted bottleneck blocks with U-Net's skip connections, which preserved fine spatial details while also capturing high-level contextual features. The ResNet50-U-Net also performed strongly, especially in delineating larger blockage regions and textured surfaces. However, its predictions often showed less precise boundaries, with minor leakage into canal walls or surrounding vegetation. InceptionV3-U-Net, by contrast, generated noticeably coarser maps, often blurring blockage boundaries and underestimating severity when debris overlapped with shadows or reflections. These limitations highlight the advantage of EfficientNetB0-U-Net for this task.

The weakest results were observed with MobileNetV2-U-Net, which frequently misclassified debris and garbage as background. While the architecture is computationally efficient and well-suited for edge devices, its reduced representational power significantly limited its performance in this task. Segmentation maps were often incomplete, failing to highlight subtle obstructions, and in many cases, the model ignored partially submerged or fine-structured debris entirely. This outcome underscores the trade-off between efficiency and accuracy: while MobileNetV2 is attractive for deployment in low-resource environments, it is not well-suited for applications requiring precise spatial resolution.

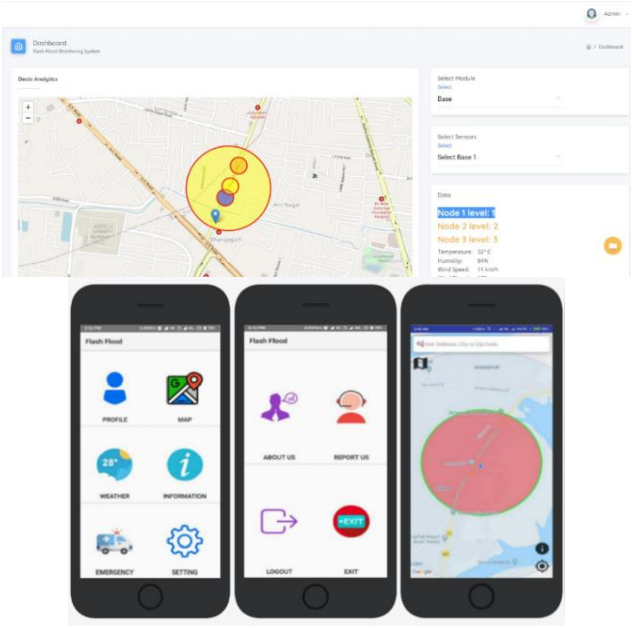


Fig. 19. Centralized dashboard (top) displaying sensor data, rainfall predictions, and visualizations, and mobile application (bottom) showing flooded areas, and emergency alerts.

A centralized dashboard and companion mobile application were developed to strengthen integration with existing flood management practices, as illustrated in Fig. 19 (dashboard at the top and mobile app at the bottom). The dashboard provides two modules: Node, showing individual sensor data, and Base Station, displaying aggregated meteorological parameters and water levels along with insights into potential flash flood events. Interactive visualizations, including a meteogram and a daily water level graph, further support decision-making. Complementing this, the mobile app integrates Google Maps to mark flooded zones, suggest alternate safe routes, and deliver emergency alerts, weather updates, and user reporting services. Together, these tools ensure that the predictive outputs of the framework can be translated into actionable information for both authorities and the community, thereby addressing the challenge

of integrating novel technologies with real-world flood management infrastructure.

V. Conclusion and Future Works

This research presented a comprehensive framework for urban flood monitoring and prediction, integrating IoT sensing, edge-based machine learning, and computer vision techniques. The system was deployed and validated in flood-prone localities of Guwahati, Assam (Anil Nagar and Zoo Road), demonstrating its ability to provide real-time rainfall forecasting, drainage monitoring, and blockage detection. The LoRa-based IoT network ensured reliable data transmission even in disaster scenarios, while the edge-deployed machine learning models, enhanced with fuzzy fusion, achieved higher prediction accuracy compared to standalone classifiers. Additionally, the use of deep learning–based semantic segmentation models enabled effective identification of blockages in drainage systems, a critical but often overlooked factor in urban flooding.

The results confirmed that the EfficientNetB0-U-Net outperformed other backbones for blockage detection, while the fuzzy fusion framework consistently improved rainfall prediction reliability. Together, these contributions advance the state of real-time, low-cost, and scalable flood monitoring systems, offering a practical tool for urban disaster preparedness in data-scarce and resource-constrained environments.

Despite these advances, several limitations remain. The dataset size for blockage detection was modest, and environmental variability (lighting, occlusion, and reflections) posed challenges to generalization. Rainfall prediction relied partly on historical datasets, which may not fully capture extreme short-duration rainfall events. Furthermore, integration of IoT networks with governance and community-level decision-making remains limited.

Although the system showed promising results, there is scope for further improvement and extension in future research.

- Expanding datasets through larger-scale image collection campaigns and crowdsourced contributions to improve the robustness of blockage detection models.
- Enhancing model generalization using advanced data augmentation, transfer learning, and multimodal fusion (combining IoT, imagery, and satellite data).
- Integration with Decision-Making Systems: Extend the framework to cover multiple urban localities with varied topographies and drainage infrastructures, alongside developing stronger links with municipal authorities and community-based early warning systems. This will ensure that model outputs not only generalize across different contexts but also translate into actionable flood preparedness measures.
- Evaluation of IoT system performance under extreme weather conditions, such as heavy rainfall and strong winds, to ensure reliability during severe flood events.

Acknowledgment

The funding and assistance from the “North Eastern Space Applications Centre (NESAC), Department of Space, Government of India, Umiam, Meghalaya” were crucial in allowing us to carry out this study, and we are very thankful to them.

References

[1] A. Gregory Jonathan N. ,Arnold,Margaret,Buys,Piet,Chen, Robert S. ,Deichmann,Uwe Klaus,Dilley, Maxx,Kjevstad, Oddvar,Lerner-Lam, Arthur L. ,Lyon, Bradfield,Yetman, “Natural disaster hotspots: A global risk analysis,” World Bank. Accessed: Jul. 02, 2025. [Online]. Available: https://documents.worldbank.org/en/publication/documents-reports/documentdetail/en/621711468175150317

[2] CRED, “EM-DAT - The international disaster database.” Accessed: Jul. 02, 2025. [Online]. Available: https://www.emdat.be/

[3] “Flood mapping and flood dynamics of the Mekong Delta: ENVISAT-ASAR-WSM based time series analyses.” Accessed: Jul. 02, 2025. [Online]. Available: https://www.researchgate.net/publication/236029131_Flood_mapping_and_flood_dynamics_of_the_Mekong_Delta_ENVISAT-ASAR-WSM_based_time_series_analyses

[4] “A comparison of selected global disaster risk assessment results | Natural Hazards.” Accessed: Jul. 02, 2025. [Online]. Available: https://link.springer.com/article/10.1007/s11069-008-9272-0

[5] K. Hansson, M. Danielson, and L. Ekenberg, “A framework for evaluation of flood management strategies,” J. Environ. Manage., vol. 86, no. 3, pp. 465–480, Feb. 2008, doi: 10.1016/j.jenvman.2006.12.037.

[6] T. Tingsanchali, “Urban flood disaster management,” Procedia Eng., vol. 32, pp. 25–37, Jan. 2012, doi: 10.1016/j.proeng.2012.01.1233.

[7] “Urban Floods in India,” ResearchGate. Accessed: Jul. 02, 2025. [Online]. Available: https://www.researchgate.net/publication/326441140_Urban_Floods_in_India

[8] “8 Dead, 1 Missing After Flash Floods in Guwahati, Assam – FloodList.” Accessed: Jul. 04, 2025. [Online]. Available: https://floodlist.com/asia/flash-floods-guwahati-assam

[9] J. D. Miller and M. Hutchins, “The impacts of urbanisation and climate change on urban flooding and urban water quality: A review of the evidence concerning the United Kingdom,” J. Hydrol. Reg. Stud., vol. 12, pp. 345–362, Aug. 2017, doi: 10.1016/j.ejrh.2017.06.006.

[10] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, “A Vision of IoT: Applications, Challenges, and Opportunities With China Perspective,” IEEE Internet Things J., vol. 1, no. 4, pp. 349–359, Aug. 2014, doi: 10.1109/JIOT.2014.2337336.

[11] D. Amaxilatis et al., “Advancing Experimentation-as-a-Service Through Urban IoT Experiments,” IEEE Internet Things J., vol. 6, no. 2, pp. 2563–2572, Apr. 2019, doi: 10.1109/JIOT.2018.2871766.

[12] Johannes Andreas Brooks, “Culvert Blockage Caused by Boulders in the Western Cape and the Development of Mitigation Measures: Physical Model Study”, [Online]. Available: https://scholar.sun.ac.za/bitstream/10019.1/108286/2/brooks_culvert_2020.pdf

[13] “A Scaled Physical Model Study of Culvert Blockage Exploring Complex Relationships Between Influential Factors: Australasian Journal of Water Resources: Vol 27 , No 1 - Get Access.” Accessed: Aug. 14, 2025. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/13241583.2021.1996679

[14] “Automating Visual Blockage Classification of Culverts with Deep Learning.” Accessed: Aug. 14, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/11/16/7561

[15] D. Eckstein, M.-L. Hutfils, and M. Winges, Global Climate Risk Index 2019. Bonn: Germanwatch, 2019.

[16] A. Patankar, “Impacts of natural disasters on households and small businesses in India,” Social Science Research Network, Rochester, NY, Dec. 23, 2019. doi: 10.2139/ssrn.3590902.

[17] H. Ali, P. Modi, and V. Mishra, “Increased flood risk in Indian sub-continent under the warming climate,” Weather Clim. Extrem., vol. 25, p. 100212, Sep. 2019, doi: 10.1016/j.wace.2019.100212.

[18] “India’s Water Crisis: Challenges, Solutions and Barriers,” ResearchGate. Accessed: Jul. 7, 2025. [Online]. Available:

https://www.researchgate.net/publication/368363874_India's_Water_Crisis_Challenges_Solutions_and_Barriers

[19] "Hydrometeorological aspects of floods in India," Natural Hazards. Accessed: Jul. 7, 2025. [Online]. Available: https://link.springer.com/article/10.1023/A:1021199714487

[20] A. K. Gupta and S. S. Nair, "Urban floods in Bangalore and Chennai: Risk management challenges and lessons for sustainable urban ecology," Curr. Sci., vol. 100, no. 11, pp. 1638–1645, 2011.

[21] H. Nguyen, M. Babel, S. Weesakul, and N. Tripathi, "An artificial neural network model for rainfall forecasting in Bangkok, Thailand," Hydrol. Earth Syst. Sci., vol. 13, pp. 1413–1425, Aug. 2009, doi: 10.5194/hess-13-1413-2009.

[22] S. H. A. Jarah, B. Zhou, R. J. Abdullah, Y. Lu, and W. Yu, "Urbanization and urban sprawl issues in city structure: A case of the Sulaymaniah Iraqi Kurdistan Region," Sustainability, vol. 11, no. 2, Art. 485, Jan. 2019, doi: 10.3390/su11020485.

[23] I. Awakimjan, "Urban flood modelling: Recommendations for Ciudad Del Plata." Accessed: Jul. 7, 2025. [Online]. Available: https://essay.utwente.nl/68990/

[24] Z. Ahmed and D. R. M. Rao, "Urban flooding – Case study of Hyderabad," Int. J. Eng. Res. Appl., vol. 2, pp. 1–6, 2013.

[25] S. G. Sabogal, N. van de Giesen, and M. ten Veldhuis, "Can urban pluvial flooding be predicted by open spatial data and weather data?," Environ. Model. Softw., vol. 85, pp. 156–171, 2016, doi: 10.1016/j.envsoft.2016.08.007.

[26] "How does imperviousness impact the urban rainfall-runoff process under various storm cases?," ResearchGate. doi: 10.1016/j.ecolind.2015.08.041.

[27] J. Abdullah and P. Y. Julien, "Distributed flood simulations on a small tropical watershed with the TREX model," Proc. Int. Conf. Hydroinformatics, 2014.

[28] J. Abdullah, N. S. Muhammad, P. Y. Julien, J. Ariffin, and A. Shafie, "Flood flow simulations and return period calculation for the Kota Tinggi watershed, Malaysia," J. Flood Risk Manag., vol. 11, no. S2, Feb. 2018, doi: 10.1111/jfr3.12256.

[29] P. T. Coulthard, R. D. Cutler, R. J. Hardcastle, and J. Hunter, "The June 2007 floods in Hull," Proc. Inst. Civ. Eng. Water Manag., vol. 161, no. 4, pp. 163–170, 2008.

[30] W. H. M. Wan Mohtar, J. Abdullah, K. N. Abdul Maulud, and N. S. Muhammad, "Urban flash flood index based on historical rainfall events," Sustain. Cities Soc., vol. 56, p. 102088, May 2020, doi: 10.1016/j.scs.2020.102088.

[31] "Flood risk and adaptation in Indian coastal cities: Recent scenarios," Appl. Water Sci.. Accessed: Jul. 8, 2025. [Online]. Available: https://link.springer.com/article/10.1007/s13201-018-0881-9

[32] S. Borah, "Assam: Guwahati gets a big boost in its fight against water-logging," EastMojo, Oct. 16, 2019. Accessed: Dec. 10, 2019. [Online]. Available: https://www.eastmojo.com/assam/2019/10/16/assam-guwahati-gets-a-big-boost-in-its-fight-against-waterlogging

[33] "Internet of Things for smart cities," ResearchGate. doi: 10.1109/JIOT.2014.2306328.

[34] "IoT driven automated object detection algorithm for urban surveillance system in smart city," ResearchGate. doi: 10.35940/ijeat.F1317.0986S319.

[35] "Data-driven solution for optimal pumping units scheduling of smart water conservancy," ResearchGate. doi: 10.1109/JIOT.2019.2963250.

[36] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," IEEE Commun. Surv. Tutor., vol. 19, no. 2, pp. 855–873, Apr. 2017, doi: 10.1109/COMST.2017.2652320.

[37] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," ICT Express, vol. 5, no. 1, pp. 1–7, Mar. 2019, doi: 10.1016/j.icte.2017.12.005.

[38] B. Arshad, R. Ogie, J. Barthelemy, B. Pradhan, N. Verstaevel, and P. Perez, "Computer vision and IoT-based sensors in flood monitoring and mapping: A systematic review," Sensors, vol. 19, no. 22, Art. 5012, Nov. 2019, doi: 10.3390/s19225012.

[39] "Regional inundation forecasting using machine learning techniques with the Internet of Things," MDPI Water. Accessed: Jul. 9, 2025. [Online]. Available: https://www.mdpi.com/2073-4441/12/6/1578

[40] "Development and application of flood control and waterlogging prevention intelligent monitoring system based on subway 'one map'," ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., vol. X-3/W1-2022, pp. 93–100, 2022. Accessed: Jul. 9, 2025. [Online]. Available: https://isprs-annals.copernicus.org/articles/X-3-W1-2022/93/2022/

[41] "floodX: Urban flash flood experiments monitored with conventional and alternative sensors," Earth Syst. Sci. Data, vol. 9, no. 2, pp. 657–666, 2017. Accessed: Jul. 9, 2025. [Online]. Available: https://essd.copernicus.org/articles/9/657/2017/

[42] Z. H. C. Soh, H. S. Gan, K. Y. Wong, W. H. Yeo, and Y. C. Lai, "Riverbank monitoring using image processing for early flood warning system via IoT," Int. J. Integr. Eng., vol. 14, no. 3, Art. 3, Jun. 2022.

[43] E. Samikwa, T. Voigt, and J. Eriksson, "Flood prediction using IoT and artificial neural networks with edge computing," in Proc. IEEE Int. Conf. Internet Things (iThings) / GreenCom / CPSCom / SmartData / Cybermatics, Nov. 2020, pp. 234–240, doi: 10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00053.

[44] A. Kumar, S. Nagar, and S. Anand, "Climate change and existential threats," in Global Climate Change, S. Singh, P. Singh, S. Rangabhashiyam, and K. K. Srivastava, Eds. Elsevier, 2021, pp. 1–31. doi: 10.1016/B978-0-12-822928-6.00005-8.

[45] "A review of the global climate change impacts, adaptation, and sustainable mitigation measures," Environ. Sci. Pollut. Res. Accessed: Jul. 12, 2025. [Online]. Available: https://link.springer.com/article/10.1007/s11356-022-19718-6

[46] K. Sami, B. A. Mohsen, K. Afef, and Z. Fouad, "Hydrological Modeling Using GIS for Mapping Flood Zones and Degree Flood Risk in Zeuss-Koutine Basin (South of Tunisia)," J. Environ. Prot., vol. 4, no. 12, Art. no. 12, Dec. 2013, doi: 10.4236/jep.2013.412161.

[47] Most. R. M. Zinat, R. Salam, M. A. Badhan, and A. R. Md. T. Islam, "Appraising drought hazard during Boro rice growing period in western Bangladesh," Int. J. Biometeorol., vol. 64, no. 10, pp. 1687–1697, Oct. 2020, doi: 10.1007/s00484-020-01949-2.

[48] Q. Yu, Y. Wang, and N. Li, "Extreme Flood Disasters: Comprehensive Impact and Assessment," Water, vol. 14, no. 8, Art. no. 8, Jan. 2022, doi: 10.3390/w14081211.

[49] B. Manandhar, S. Cui, L. Wang, and S. Shrestha, "Urban Flood Hazard Assessment and Management Practices in South Asia: A Review," Land, vol. 12, no. 3, Art. no. 3, Mar. 2023, doi: 10.3390/land12030627.

[50] F. M. Underwood, "Describing long-term trends in precipitation using generalized additive models," J. Hydrol., vol. 364, no. 3–4, Art. no. 3–4, 2009.

[51] "Distributions of Annual Maximum Rainfall Series of North-East India | Request PDF." Accessed: Jul. 23, 2025. [Online]. Available: https://www.researchgate.net/publication/279195087_Distributions_of_Annual_Maximum_Rainfall_Series_of_North-East_India

[52] "Generalized Additive Models | 7 | Statistical Models in S | Trevor J." Accessed: Jul. 23, 2025. [Online]. Available: https://www.taylorfrancis.com/chapters/edit/10.1201/9780203738535-7/generalized-additive-models-trevor-hastie

[53] "The Unquiet River: A Biography of the Brahmaputra | Oxford Academic." Accessed: Jul. 23, 2025. [Online]. Available: https://academic.oup.com/book/36968

[54] "A Historical Understanding of Assam's Floods | Economic and Political Weekly." Accessed: Jul. 23, 2025. [Online]. Available: https://www.epw.in/engage/article/historical-understanding-assams-floods

[55] N. Jamwal, "Examining Assam's disaster readiness after the 2022 floods," Dialogue Earth. Accessed: Jul. 23, 2025. [Online]. Available: https://dialogue.earth/en/climate/examining-assams-disaster-readiness-after-the-2022-floods/

[56] "Majuli Island and Assam's Rivers Face Unpredictable Flooding in 2024." Accessed: Jul. 23, 2025. [Online]. Available: https://www.downtoearth.org.in/natural-disasters/assam-floods-2024-unprecedented-timing-and-fury-grips-state

[57] "What's really behind Assam's worsening floods? | PreventionWeb." Accessed: Jul. 23, 2025. [Online]. Available: https://www.preventionweb.net/news/india-whats-really-behind-assams-worsening-floods

[58] "Sixth Assessment Report — IPCC." Accessed: Jul. 23, 2025. [Online]. Available: https://www.ipcc.ch/assessment-report/ar6/

[59] "Multi-Model Ensemble Prediction of Summer Precipitation in China Based on Machine Learning Algorithms." Accessed: Jul. 25, 2025. [Online]. Available: https://www.mdpi.com/2073-4433/13/9/1424

[60] C. Z. Basha, N. Bhavana, P. Bhavya, and S. V, "Rainfall Prediction using Machine Learning & Deep Learning Techniques," in 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Jul. 2020, pp. 92–97. doi: 10.1109/ICESC48915.2020.9155896.

[61] "Study of short term rain forecasting using machine learning based approach | Request PDF." Accessed: Jul. 25, 2025. [Online]. Available: https://www.researchgate.net/publication/336831403_Study_of_short_term_rain_forecasting_using_machine_learning_based_approach

[62] "A novel approach for precipitation forecast via improved K -nearest neighbor algorithm | Request PDF," ResearchGate, doi: 10.1016/j.aei.2017.05.003.

[63] "Precipitation Forecasting in Northern Bangladesh Using a Hybrid Machine Learning Model." Accessed: Jul. 25, 2025. [Online]. Available: https://www.mdpi.com/2071-1050/14/5/2663

[64] K. Ahmed, D. A. Sachindra, S. Shahid, Z. Iqbal, N. Nawaz, and N. Khan, "Multi-model ensemble predictions of precipitation and temperature using machine learning algorithms," Atmospheric Res., vol. 236, p. 104806, May 2020, doi: 10.1016/j.atmosres.2019.104806.

[65] "Machine Learning for Climate Precipitation Prediction Modeling over South America." Accessed: Jul. 25, 2025. [Online]. Available: https://www.mdpi.com/2072-4292/13/13/2468

[66] Y. Zhang and A. Ye, "Machine Learning for Precipitation Forecasts Postprocessing: Multimodel Comparison and Experimental Investigation," Nov. 2021, doi: 10.1175/JHM-D-21-0096.1.

[67] "Improved Numerical Weather Prediction Using IoT and Machine Learning | Request PDF," in ResearchGate. doi: 10.1007/978-981-19-8086-2_109.

[68] P. Shah, Y. A, U. Khaitan, and S. Kayalvizhi, "Weather Management System Using Machine Learning Algorithm And IOT," in 2023 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence (RAEEUCCI), Apr. 2023, pp. 1–4. doi: 10.1109/RAEEUCCI57140.2023.10134454.

[69] "Rainfall Forecasting System Using Machine Learning Technique and IoT Technology for a Localized Region." Accessed: Jul. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/366777278_Rainfall_Forecasting_System_Using_Machine_Learning_Technique_and_IoT_Technology_for_a_Localized_Region

[70] "Real-Time Rainfall Prediction System Using IoT and Machine Learning | SpringerLink." Accessed: Jul. 26, 2025. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-031-23973-1_10

[71] Y. Xu et al., "Research on particle swarm optimization in LSTM neural networks for rainfall-runoff simulation," J. Hydrol., vol. 608, p. 127553, May 2022, doi: 10.1016/j.jhydrol.2022.127553.

[72] "Artificial Neural Network Models for Rainfall Prediction | European Journal of Electrical Engineering and Computer Science." Accessed: Jul. 26, 2025. [Online]. Available: https://ejece.org/index.php/ejece/article/view/313

[73] "An IoT-Based Predictive Analytics for Estimation of Rainfall for Irrigation." Accessed: Jul. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/343656881_An_IoT-Based_Predictive_Analytics_for_Estimation_of_Rainfall_for_Irrigation

[74] "An Intelligent Weather Prediction System Based on IOT | IEEE Conference Publication | IEEE Xplore." Accessed: Jul. 26, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/9455883

[75] "Low Cost IoT based Flood Monitoring System Using Machine Learning and Neural Networks: Flood Alerting and Rainfall Prediction | IEEE Conference Publication | IEEE Xplore." Accessed: Jul. 26, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/9074928

[76] Ali, M. H. M., Asmai, S. A., Abidin, Z. Z., Abas, Z. A., & Emran, N. A. (2022). Flood prediction using deep learning models. International Journal of Advanced Computer Science and Applications (IJACSA), 13(9), 972–981.

[77] "Real-Time Rain Prediction in Agriculture using AI and IoT: A Bi-Directional LSTM Approach - Google Search." Accessed: Jul. 26, 2025. [Online]. Available: https://www.google.com/search?q=Real-Time+Rain+Prediction+in+Agriculture+using+AI+and+IoT%3A+A+Bi-Directional+LSTM+Approach&oq=Real-Time+Rain+Prediction+in+Agriculture+using+AI+and+IoT%3A+A+Bi-Directional+LSTM+Approach&gs_lcrp=EgZjaHJvbWUyBggAEEUYOTIGCAEQRRg8MgYIAhBFGDzSAQkxNDA0ajBqMTWoAgiwAgHxBT2T-laKvkmb&sourceid=chrome&ie=UTF-8

[78] "Development of a Secured IoT-Based Flood Monitoring and Forecasting System Using Genetic-Algorithm-Based Neuro-Fuzzy Network." Accessed: Jul. 26, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/13/3885

[79] "CIRIA C689 Culvert design and operation guide," ResearchGate. Accessed: Aug. 14, 2025. [Online]. Available: https://www.researchgate.net/publication/370033204_CIRIA_C689_Culvert_design_and_operation_guide

[80] Marouane, E. M. (2021). Towards a Real Time Distributed Flood Early Warning System. International Journal of Advanced Computer Science and Applications (IJACSA), 12(1), 34–41.

[81] "A physical model study of culvert blockage by large urban debris," ResearchGate. Accessed: Aug. 14, 2025. [Online]. Available: https://www.researchgate.net/publication/293637298_A_physical_model_study_of_culvert_blockage_by_large_urban_debris

[82] Md Rashid, N. A., Abidin, Z. Z., & Abas, Z. A. (2024). Integrating Multi-Agent System and Case-Based Reasoning for Flood Early Warning and Response System. International Journal of Advanced Computer Science and Applications (IJACSA), 15(12), 112–120.

[83] Bakhsh, S. T., Basheri, M., Ahmed, N., & Shahzad, B. (2020). A flood forecasting model based on wireless sensor and actor networks. International Journal of Advanced Computer Science and Applications (IJACSA), 11(5), 438–446.

[84] "Regression on Deep Visual Features using Artificial Neural Networks (ANNs) to Predict Hydraulic Blockage at Culverts," ResearchGate. Accessed: Aug. 24, 2025. [Online]. Available: https://www.researchgate.net/publication/351448845_Regression_on_Deep_Visual_Features_using_Artificial_Neural_Networks_ANNs_to_Predict_Hydraulic_Blockage_at_Culverts

[85] R. R. Patil, R. K. Calay, M. Y. Mustafa, and S. M. Ansari, "AI-Driven High-Precision Model for Blockage Detection in Urban Wastewater Systems," Electronics, vol. 12, no. 17, p. 3606, Jan. 2023, doi: 10.3390/electronics12173606.

[86] R. Alshawi et al., "SHARP-Net: A Refined Pyramid Network for Deficiency Segmentation in Culverts and Sewer Pipes," Aug. 02, 2024, arXiv: arXiv:2408.08879. doi: 10.48550/arXiv.2408.08879.

[87] R. Vandaele, S. L. Dance, and V. Ojha, "Deep learning for automated trash screen blockage detection using cameras: Actionable information for flood risk management," J. Hydroinformatics, vol. 26, no. 4, pp. 889–903, Apr. 2024, doi: 10.2166/hydro.2024.013.

[88] "CCTV image ‐ based classification of blocked trash screens - Smith - 2025 - Journal of Flood Risk Management - Wiley Online Library." Accessed: Aug. 24, 2025. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/jfr3.13038

[89] Muniandy, B., Maidin, S. S., Batumalay, M., & Dhandapani, L. (2025). Flood Prevention System Using IoT. International Journal of Advanced Computer Science and Applications (IJACSA), 16(3), 77–85.

Privacy-Aware ML Framework for Dynamic Query Formation in Multi-Dimensional Data

B Bhavani[1], Dr. Haritha Donavalli[2]
Research Scholar, Department of Computer Science and Engineering, K L (deemed to be university), Andhra Pradesh, India [1]
Professor, Department of Computer Science and Engineering, K L (deemed to be university), Andhra Pradesh, India [2]

***Abstract*—Interactive data exploration at scale remains constrained by 1) weak adaptability to shifting query workloads, 2) limited and post hoc error guarantees, 3) poor scalability under dynamic, high-dimensional data, 4) sparse user guidance during query formulation, and 5) non-trivial system overheads from learned or probabilistic components. We propose an end-to-end, privacy-aware framework that dynamically forms SQL queries for multi-dimensional data using randomized signals derived from personal web usage. The method integrates: 1) on-device user modeling that converts browsing interactions into preference embeddings under local differential privacy; 2) a constrained-randomization layer that enforces coverage and diversity to avoid filter bubbles while remaining responsive to user intent; 3) a contextual bandit policy (with optional deep reinforcement learning extension) that selects or completes query templates using signals from user profiles, session context, and data synopses; and 4) an error-aware AQP executor combining stratified/pilot sampling, synopsis reuse, and confidence-interval gating with automatic sample escalation. This design directly addresses the above limitations: the bandit adapts online to workload shifts; the AQP layer provides pre-execution feasibility checks and per-query error control; synopsis reuse and AB-tree–style random sampling maintain low latency under updates; and a guidance module (predictive autocompletion with information-gain scoring) reduces user effort while preserving exploration diversity. To evaluate effectiveness, we introduce a privacy-preserving training regimen (federated updates over DP-noised profiles) and a novel benchmark protocol measuring time-to-insight, error compliance under differential privacy, session diversity, and latency against strong baselines. The result is an ML-driven exploration loop that achieves error-bounded interactivity, robust personalization, and scalable performance on evolving, high-dimensional datasets, while providing evaluation metrics that capture both user experience and privacy-preserving guarantees.**

***Keywords*—Dynamic query formation; Approximate Query Processing (AQP); local differential privacy; contextual bandits; reinforcement learning; constrained randomization; multi-dimensional data exploration**

I. Introduction

Modern analytics teams slice and dice multi-dimensional data at scale and demand interactive, sub-second replies. Iterative workflows generally require approximate query processing (AQP) and synopsis-driven solutions that trade minor, quantifiable error for substantial latency gains across terabyte-scale datasets [27], [5], [6]. In parallel, research examines how to lead users during exploration—through recommendations, active learning, and result diversification—to maximize understanding rather than repeated or empty outcomes [24], [25], [26], [14], [13].

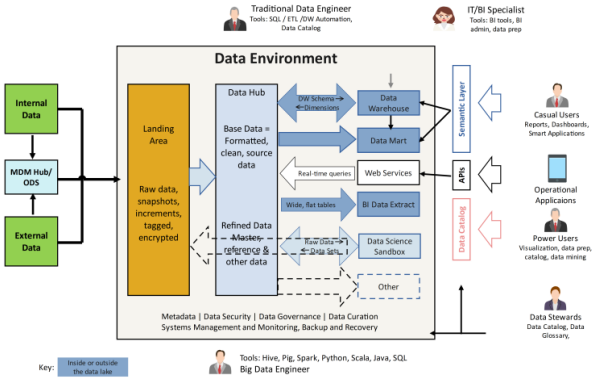


Fig. 1. General architecture of web usage data exploration.

Fig. 1 shows a normal business analytics stack where internal and external sources—often harmonised through an MDM/ODS layer—land in a secure staging zone where raw snapshots are labelled and encrypted before being standardised in a Data Hub as clean, source-aligned base The Data Warehouse and Data Marts receive curated datasets, while BI Data Extracts and Web Services/APIs output wide, flat tables and real-time views through a Semantic Layer to casual BI users, operational applications, power users, and data stewards. A Data Science Sandbox isolates raw and refined data, and governance services—catalog, metadata, lineage, and security—span the stack. Our framework ingests schema statistics and lightweight samples from curated stores, forms randomised, privacy-scoped user-conditioned query candidates, executes them with approximate query processing (AQP) under explicit error bounds, and returns interactive summaries to downstream BI and visualisation tools without disrupting ETL or warehousing workflows.

Despite advances, the state of the art has five limitations. Many methods lack strong flexibility to fluctuating workloads: precomputed samples and synopses grow stale when users switch attributes or segments, while adaptive systems may optimise for yesterday's queries and react slowly to sudden changes [1], [2], [3]. Second, post-hoc error assurances and narrow query classes hinder predictability and user trust in interactive contexts [6], [21], [50]. Third, dynamic, high-dimensional datasets have scalability issues—managing numerous stratified samples or multi-attribute indexes occupies

space and requires maintenance [11], [23], [7]. Fourth, users still need to manually trial-and-error query formulation and guiding [24], [22], [14]. Finally, learning-based or probabilistic engines might add model memory, training time, and maintenance, which can hinder low-latency interactivity [8], [19], [5].

Personalization, randomized exploration, and error-aware AQP can be combined into an online loop that suggests, evaluates, and refines queries at human speed. Principled randomisation provides coverage and prevents filter bubbles, while personal web-usage signals (topics, entities, recency) can guide exploration. Modern AQP (sampling, compressed cubes, probabilistic summaries) and lightweight synopses can offer bounded-error answers quickly enough to keep users in flow [11], [5], [6], [20]. Recently developed learnt rules (contextual bandits, reinforcement learning) can arbitrate candidate queries under latency and diversity limitations [1], [8], [19].

1) Challenges: Designing a machine–learning framework for dynamic query creation over multi-dimensional data with randomised, privacy-scoped personalisation presents many system and model problems: 1) Data heterogeneity and schema drift: attributes vary in type, granularity, and nomenclature; sources fluctuate, breaking learnt mappings and cached synopses. 2) High dimensionality and combinatorial explosion: attribute-value predicates and group-bys increase exponentially, requiring rigorous pruning, templating, and coverage guarantees. 3) Latency-accuracy trade-off: interaction needs sub-second medians while respecting statistical error limitations; tight SLAs make balancing sample size, synopsis reuse, and escalation difficult. 4) Sample bias and uncertainty quantification: stratification, skew, and infrequent segments endanger unbiased results; confidence intervals and variance models must survive reuse and data updates [27]. 5) Privacy constraints [21]: local differential privacy decreases signal quality; balancing value and protection, and propagating noise through selection and estimation is difficult. 6) User modelling is complicated by cold start, idea drift, and session volatility; signals are scarce, implicit, and noisy. 7) Explore--exploit control: randomised candidate generation requires coverage guarantees to avoid myopic loops and online policy adaptation without quality degradation. 8) Reward shaping and counterfactual tracking are needed to ensure feedback reliability: presentation bias and interface effects complicate clicks, dwell, and refinements. 9) Interpretability and trust: questions, error bars, and trade-offs must be explained to consumers to accept approximate answers. 10) Scalability and resource efficiency: big, frequently updated tables strain CPU, memory, and storage budgets for sketches, samples, and caches. 11) Streaming and freshness: incremental synopses and confidence assurances for arrivals/out-of-order data are difficult. 12) Robustness and fairness: regulations should not disregard minorities, withstand hostile or inadvertent outliers, and remain consistent across workloads. 13) Reproducibility and governance: randomised selection hinders auditability; queries, samples, and models must be seedable with lineage. 14) Benchmarking and evaluation: suites must examine latency, CI compliance, coverage/diversity, and time-to-insight because no single metric covers utility. 15) Integration and portability: interfaces must work with catalogues, BI tools, and engines without changing ETL/ELT pathways and across approximate executors.

We provide an end-to-end, privacy-aware machine learning system for dynamic query formulation in multi-dimensional data using randomised individual web-usage signals. Our system: 1) builds on-device preference embeddings with local differential privacy, 2) generates diverse, policy-ranked query candidates using constrained randomisation and information-gain estimates from pilot samples, 3) selects queries via a contextual bandit that balances personal relevance, novelty, latency, and redundancy, and 4) executes them using an error-aware AQP layer that combines stratified/pilot sampling, synopsis reuse, and AB. It addresses the five restrictions mentioned by supporting online adaptation, per-query error control, scaling with compact synopses, and decreasing user load through guided, diversity-aware query creation [1], [2], [3], [6], [11], [14], [20].

2) Our contribution: The study provides four contributions: A personalized, privacy-aware modelling pipeline that seeds query intent and reduces cold-start by converting web-usage events into exploratory preferences. A constrained-randomization approach with coverage guarantees and information-gain scoring that preserves exploration diversity and relevance. A contextual bandit strategy with an RL extension that optimizes user engagement, information gain, latency, and redundancy under per-query cost and error limitations. An error-aware AQP executor that increases sample sizes and reuses synopses/materialized summaries for bounded-error results at interactive latencies [6], [11], [12], [5].

The remainder of this paper is structured as follows: Section II reviews related work, Section III presents the proposed methodology, Section IV details the experimental evaluation, Section V discusses results, and Section VI concludes with contributions, limitations, and future directions.

II. Related Work

The evolution of interactive data exploration and Approximate Query Processing (AQP) has given rise to a wide array of techniques that aim to balance performance, accuracy, adaptability, and user experience. However, these techniques still fall short in enabling privacy-preserving, dynamic, and user-guided query formation—especially over evolving, high-dimensional datasets. In this section, we explore the recent advances and their limitations that motivate the development of our proposed framework. The detailed work is shown in Table I.

Zhang et al. [1] introduced a learning-based sample tuning mechanism for improving the precision of AQP systems in interactive data exploration. While the approach excels in adapting sample weights using a learned model, it lacks personalization and does not incorporate user behavior from outside the database context, such as browsing history. Similarly, Engelmann et al. [2] proposed AQP-Reuse, which reuses intermediate approximate query results to accelerate

future interactions. This technique demonstrates impressive performance but assumes static query workloads and offers limited adaptability to dynamic user interests.

TABLE I. RELATED WORK COMPARISON

Reference	Key Contribution	Approach	Limitation Addressed in Proposed Work
Zhang et al.[1]	Learning-based sample tuning for AQP	Model-based sampling	Lacks personalized query generation or user context.
Engelmann et al. [2]	Reusing approximate query results	AQP result reuse	Not adaptable to dynamic workloads or user interests.
Li et al. [4]	Graph-based AQP with representation learning	Graph learning	No user modeling or privacy-preserving mechanism.
Zhao et al. [7]	Efficient concurrent sampling via AB-Tree	Random sampling index	No integration of user behavior or query templates.
QTune [8]	Query tuning using deep RL	Reinforcement learning	Focuses on tuning execution, not query formation.
Shen et al.[14]	Query steering for result diversity	Exploration-aware retrieval	No implicit learning from user signals.
He et al. [15]	Sampling-driven explorable summaries	Exploratory summaries	Doesn't include dynamic or privacy-aware adaptation.
Lin et al. [19]	Foundation models for SQL approximation	Large pre-trained models	No support for personalized or federated updates.
Verbruggen et al. [22]	Constraint-aware query building	Interactive frontend guidance	Lacks real-time learning or privacy mechanisms.
Dimitriadou et al. [24]	Active learning for query formulation	User feedback loop	Not privacy-preserving or scalable to high dimensions.

Maroulis et al. [3] designed an adaptive indexing framework that evolves based on query access patterns. Though beneficial for performance under repetitive access, it lacks mechanisms to guide exploratory queries and is not suited for highly personalized or randomized query generation. Li et al. [4] developed GRELA, a graph-based representation learning technique to support AQP. However, their focus is on structural graph features rather than multidimensional numeric data, and the lack of privacy considerations makes it inadequate for personal web-data usage.

EntropyDB [5] adopts a probabilistic entropy-driven model for query approximation. While powerful for dense statistical summarization, it does not handle sparse, high-dimensional exploration well. Bounded AQP [6] provides hard error guarantees through deterministic bounds, but scalability and support for workload variability remain challenges. Zhao et al. [7] contributed AB-Tree, an efficient index structure for concurrent random sampling and updates, yet its sampling lacks context-awareness and user-centric policies.

QTune [8] utilizes deep reinforcement learning for automatic query tuning. Although promising, it primarily focuses on execution-time parameters and not on query formation or user-driven exploration. In contrast, Rimi et al. [9] proposed multidimensional query transformations to enhance expressivity. Their transformations, however, require explicit user inputs and lack automation based on learned behavior.

Nguyen et al. [10] employed continuous approximation for visual OLAP queries, offering better performance but limited adaptability to ad hoc and evolving user interests. Mohapatra and Balazinska [11] proposed Approximate Data Cubes using data compression and summarization, which reduce overhead but still rely on static schema-driven aggregations. Roh et al. [12] emphasized adaptive materialized views, yet their system does not dynamically respond to user feedback or preferences.

Large et al. [13] explored predictive sampling for query autocompletion. While enhancing interactivity, it falls short in promoting exploration diversity and privacy-preserving behavior modeling. Shen et al. [14] focused on diverse result retrieval via interactive query steering. Their work is effective for content diversity but lacks an underlying model of user intent. He et al. [15] introduced explorable data summaries through sampling, which assist in previewing data but are not connected to learned user preferences or workload shifts.

Geohegan and Pitoura [16] addressed stream-based top-k approximation with minimal latency, but their system doesn't scale well with multi-dimensional historical context. Müller et al. [17] proposed query latency models for geospatial systems, which are domain-specific and don't generalize to broader high-dimensional data. Fu et al. [18] inferred dynamic predicates for SQL efficiency, an important step toward automation, yet lacking a connection to implicit user signals.

Lin et al. [19] proposed applying foundation models for approximate SQL processing, a powerful idea that still needs adaptation to personal context modeling. Farid et al. [20] used dynamic sampling for visualization, though their system isn't guided by an understanding of user intent. Symeonidis et al. [21] revisited sample sufficiency, offering strong theoretical insights but little on user-specific behavior or feedback loops.

Verbruggen et al. [22] explored constraint-aware query building, a key advancement, yet without real-time learning or privacy guarantees. Das et al. [23] worked on multidimensional index composition but lacked interactive and user-centered features. Dimitriadou et al. [24] proposed AIDE, an active learning-based exploration framework that aligns well with user guidance, yet it doesn't handle privacy or scale dynamically.

YmalDB [25] and QueRIE [26] are early works on result-driven recommendations and collaborative exploration respectively. Though foundational, they are not scalable or privacy-preserving. Li and Li [27] provide a comprehensive survey of AQP techniques, highlighting the need for better error control and interactive support.

Foundational concepts like Data Cube [28], Quotient Cube [29], Count-Min Sketch [30], and HyperLogLog [31] offer statistical underpinnings for AQP but were not designed with

modern interactivity, user context, or privacy in mind. Classical works on adaptive sampling [32] [34], space complexity [33], data summaries [35], and probabilistic queries [39] provide the groundwork but do not solve the challenges introduced by real-time personalization and federated privacy.

Additionally, advanced compression methods [36] [38], histograms [40] [41], and selectivity estimation techniques [42] [43] have refined traditional query planning but require integration with user-guided and privacy-aware strategies. Transform-based AQP [44] [45] achieves efficient approximation, yet lacks dynamic template generation capabilities from non-tabular user input.

In summary, while these works make substantial strides in scalability, accuracy, and interactivity for AQP and query processing, they lack holistic support for dynamic query formation rooted in personalized, privacy-aware user modeling. Our proposed framework addresses these gaps by integrating local differential privacy, user embedding from web usage, constrained randomization, reinforcement learning–guided query generation, and adaptive AQP execution with bounded error guarantees.

III. Methodology

In this section, we discuss on proposed method and its modules. The existing work diagram in Fig. 2 illustrates a traditional metadata-enriched recommendation model used for predicting candidate ratings R_{j_k} between users and items. In this architecture, each user u_j and item i_k is associated with a set of auxiliary attributes that capture contextual and descriptive features. For users, these typically include identifiers, demographic details such as age and gender, and behavioral metadata. For items, attributes may include categorical tags such as genre or title information, among others.

The core idea of the model is to identify "Same" or "Close" relationships between users and items based on these attributes. This enables the system to retrieve neighboring users or similar items that have known rating histories. From this information, two sub-models—user-oriented $R^u_{y,\gamma}$ and item-oriented R^I_x,k rating predictors—are engaged to extrapolate the likely rating score for the current candidate user-item pair. The prediction R_{jk} is generated by aggregating or interpolating this information.

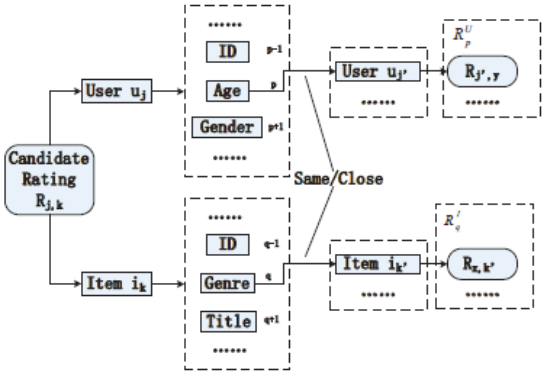


Fig. 2. Overview of existing recommendation system diagram.

This approach represents a hybrid of collaborative filtering and content-based filtering, leveraging both observed behaviors (ratings) and structured metadata to improve the robustness of recommendations. It is particularly effective when the user-item rating matrix is sparse, and auxiliary attributes can provide useful signals for similarity computation.

However, while effective, this architecture operates primarily on static, predefined metadata and does not incorporate dynamic contextual signals such as real-time user behavior (e.g. clickstream or browsing data). Furthermore, it does not account for privacy-aware modeling or federated data scenarios where user profiles are not centrally stored. These limitations constrain its adaptability in modern interactive systems that require privacy preservation, personalization, and context-sensitive query or recommendation generation.

In contrast, the proposed machine learning framework shown in Fig. 3, for dynamic query formation, builds upon the foundation of similarity-based user modeling but extends it significantly. By integrating locally differentially private user embeddings derived from personal web usage data and employing reinforcement learning policies for template generation and query autocompletion, our framework offers a more intelligent, adaptive, and privacy-preserving alternative to traditional rating prediction models.

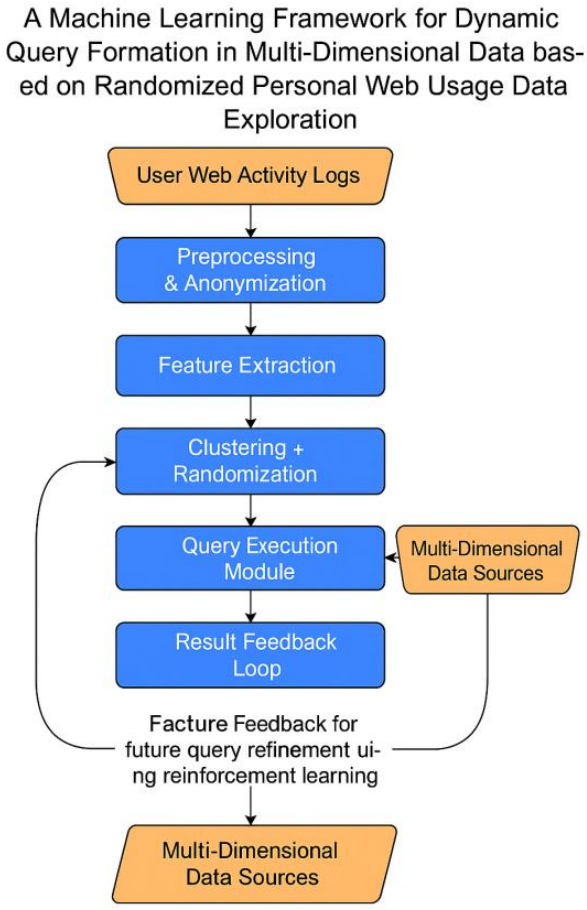


Fig. 3. Overview of the proposed architecture.

Let $\mathfrak{D}$ denote one or more multi-dimensional data sources with attributes $\mathfrak{A} = \{A_1,\ldots,A_d\}$ and numeric measures $\mathfrak{M}$. At interaction step t, the system forms a SQL query q_t from templates (filters P, group-bys g, aggregate $\varphi \in \{SUM, AVG, COUNT\}$) and executes it with approximate query processing (AQP) to maintain interactivity. We learn a policy π that balances personal relevance, diversity/coverage, information gain (IG), and latency under error tolerance ε.

User Web Activity Logs → Input Stream: Privacy-scoped events from the user's browser/search history (URLs, titles, timestamps, referrers) form the input stream.

- Sessionization: split events by inactivity gap (e.g., 30 minutes) into sessions $S = \{S_1,\ldots\}$.
- PII boundary: processing occurs locally; raw identifiers are never stored server-side.

Preprocessing and Anonymization:

1) Canonicalization: strip tracking query parameters (keep a whitelist), lowercase host/path, unshorten URLs.

2) PII removal: redact emails, phone numbers, IDs via regex; hash user IDs with per-device salt.

3) Tokenization: combine title + snippet → tokens; keep top-K TF-IDF terms per session.

4) Local DP noise: add Gaussian/Laplace noise to term counts and later to profile vectors (privacy budget ε_{DP}).

Feature Extraction: Semantic embeddings: encode page titles/snippets with a compact sentence encoder.

Schema alignment: Build attribute/value lexicons from $\mathfrak{D}$ (column names, value samples). Cross-encode e_{sess} with attribute prompts to obtain attribute preference scores $w \in \mathbb{R}^d$ (softmax-normalized), then apply DP noise to get w~.

User profile: Maintain a running profile $u \leftarrow \alpha \cdot u + (1-\alpha) \cdot e_{sess}$ (EMA) and sparse interest tags (top terms/entities) for interpretability.

Clustering + Randomization: Keep K-means (or incremental DP-K-means) over recent session embeddings to discover interest clusters $C = \{c^1,\ldots,c_K\}$. For each cluster c, store centroid μ_c, supported attributes $G_c \subseteq \mathfrak{A}$ (highest scoring by w~), and exemplar values V_c (frequent/representative values from logs and data samples).

Coverage-aware randomized selection: cluster score at time t.

$$s_c = (1-\lambda) \cdot \mathrm{sim}(u, \mu_c) + \lambda \cdot \mathrm{IG}_c - \eta \cdot \mathrm{red}_c$$

Draw $c_t \sim \mathrm{softmax}(\frac{s_c}{T})$ while enforcing minimum coverage over attributes/values within a session (reject over-used clusters).

Candidate generation: from c_t, produce K candidate queries by filling templates: • Drill-down (GROUP BY $g \in G_c$ with predicates on top-k values from V_c); • Slice-and-dice (2D group-by pairs with top-k filters); • Time roll-ups (windowed aggregates where time exists); • Top-k segments (rank by estimated lift or variance).

Query Execution Module (Approximate Query Processing):

Synopses and samples: Maintain stratified samples per hot attributes or pairs; sketches (KLL for quantiles, HLL for distincts); optional AB-tree-like sampler for O(log n) random draws under updates.

Feasibility and cost: for each candidate q, on a small pilot sample S_0 estimate variance σ^2, selectivity, and latency using a learned regressor. Accept if predicted CI width $\leq \varepsilon$; else escalate sample size or fall back to exact execution if under time budget.

Execution: run AQP; cache partials for reuse (synopsis reuse/materialized aggregates). Emit ($\hat{y}$, CI, latency).

Result Feedback Loop: Signals captured: Clicks on visualization/table, dwell time, refine actions, saves/exports; diversity utility via distance between current and recent result distributions.

Reward shaping: $r_t = \alpha \cdot u_t + \beta \cdot \mathrm{IG}(q_t) - \gamma \cdot \widehat{\mathrm{lat}}(q_t) - \delta \cdot \mathrm{red}(q_t)$, where u_t aggregates interaction signals.

Profile updates: Online updates of u and w~; increment coverage counters.

Reinforcement Learning for Future Refinement

State: s_t = [u, w~, e_{sess}, synopsis stats, history embedding].

Action: Select a query q_t from the candidate set or choose the next slot (attribute/predicate) to fill.

Policy: Begin with a contextual bandit (LinUCB or Thompson Sampling) for robust online learning: $\hat{r}(q) = \theta^T \cdot \phi(s_t, q)$ with UCB/TS exploration. Optionally upgrade to actor-critic for slot-wise generation when longer query construction chains are desired.

Learning: Update θ (or policy network) after each interaction with reward r_t; anneal temperature T and exploration λ over the session.

Multi-Dimensional Data Sources ⇄ System Integration:

Connectors: SQL engines, data warehouses, data lakes. Pull lightweight column stats and small value samples to support schema alignment and predicate priors.

Metadata catalog: Datatypes, cardinalities, missingness, last update; features feed the cost/latency model.

Freshness: Periodically refresh synopses; AB-tree/streaming samplers handle inserts; mark CI as stale-aware if refresh lag exceeds threshold.

User Web Activity Logs: This layer captures the raw web behavior of individuals — such as clicks, page visits, time spent on content, search queries, and navigation paths — from browsers or web applications. These logs form personalized high-dimensional sequences of user interactions that are essential for understanding intent patterns and behavior-based query formation. This step also includes secure logging and privacy-aware data tagging.

Preprocessing and Anonymization: The preprocessing stage performs essential data cleaning, transformation, and structuring, while the anonymization component ensures

privacy preservation: Data Cleaning: Removes null, redundant, or irrelevant values. Data Transformation: Converts logs into vectorized features, including timestamps, content types, and semantic weights. Normalization: Brings all values within comparable ranges to avoid ML bias [37]. Generalization: Abstracts specific values (e.g. URLs → Categories). Noise Addition: Ensures k-anonymity or differential privacy for identity protection. Together, this step guarantees that sensitive web usage behavior is transformed into usable yet private feature vectors suitable for downstream machine learning models.

Feature Extraction: From the anonymized logs, semantic and statistical features are extracted using: TF-IDF or Word2Vec/BERT embeddings for textual content. Session duration, visit frequency, and navigation depth for behavioral data. Temporal sequences to model contextual intent. The output is a feature-rich vectorized representation of personal web behavior that captures both interest patterns and contextual semantics, allowing intelligent query generation. Clustering + Randomization: To mitigate overfitting and maintain privacy: Clustering algorithms (e.g., K-Means, DBSCAN) group similar user behavior patterns. Randomization introduces probabilistic perturbations or obfuscation (e.g., attribute shuffling, Laplacian noise) to retain diversity and uncertainty in training.

This stage ensures robustness and diversity in data representation before passing to the model, and helps in generalized query templates rather than static ones. Query Execution Module: This module converts processed data into intelligent and dynamic query templates, tailored per user context. Key ML components include: Classification/Prediction models (e.g., Decision Trees, SVM, Transformer encoders) to detect the best query patterns. Intent prediction models to understand likely user goals. Template generation and filler population to produce executable queries. Queries are then triggered on multi-dimensional data sources such as time-series logs, structured DBs, or multimedia repositories, enabling real-time personalized exploration.

Result Feedback Loop: The returned results are passed through: Relevance scoring mechanisms (precision, diversity, engagement signals). User interaction feedback (e.g., click-through rate, dwell time). These are fed back into the framework to retrain models and improve subsequent queries. A reinforcement learning loop (e.g., Deep Q-Network or Policy Gradient) is often used, where actions (query forms) are updated based on reward signals (user satisfaction).

Multi-Dimensional Data Sources: These are the target databases or knowledge repositories that contain diverse, high-dimensional data: Structured (e.g., SQL-based), Semi-structured (e.g., JSON/XML web logs), Unstructured (e.g., text, images, videos). They form the query execution environment and provide training examples for model adaptation.

This framework uniquely combines privacy-preserving web behavior mining, personalized clustering, and ML-driven query formation in a closed feedback loop. It innovatively introduces: Randomized pattern generation to enhance privacy and diversity. Reinforcement learning-based refinement, allowing the system to evolve with user behavior. Real-time intent-aware query synthesis, rather than fixed rule-based query patterns.

IV. Results

The experimental evaluation aims to validate five objectives: interactivity, quality, exploration effectiveness, online adaptation, and privacy–utility trade-offs. Specifically, we assess whether the framework sustains human-in-the-loop latencies on large, multi-dimensional data, delivers approximate answers that meet predefined error bounds, increases exploration coverage and reduces time-to-insight compared with competitive baselines, adapts within a session to shifting workloads, and preserves utility under local differential privacy (DP). Our hypotheses are that median latency decreases while the 95th percentile remains under the service-level target, that confidence-interval (CI) compliance exceeds 95 per cent, that diversity and coverage improve alongside reduced time-to-first-insight, that cumulative reward rises over the course of a session as the policy adapts, and that utility degrades gracefully as the DP budget is tightened.

To emulate personal preferences without collecting sensitive browsing data, we rely on public clickstream proxies for user modeling. News-oriented sessions (e.g., MIND) and commerce-oriented sessions (e.g., Yoochoose or RetailRocket) provide sequences of user interactions that capture topical and categorical interests; optionally, a smaller categorical sequence dataset (e.g., the UCI MSNBC corpus) is included for additional variety. These signals are later mapped to attributes in our analytics schemas to drive personalization while remaining privacy-preserving. For query execution, we evaluate on three representative domains. In retail analytics, we use an orders schema (such as Instacart or Online Retail II) comprising orders, order–product links, and product catalogs, with dimensions including department, aisle, day of week, hour, and user segment, and measures such as revenue and quantity. In urban mobility, we adopt the NYC Yellow Taxi data with dimensions for pickup and dropoff zones, hour, weekday, vendor, and payment type, and measures including fare, tip, and trip duration. In media analytics, we use IMDb or MovieLens-20M with dimensions for genre, year, region, and language, and measures for average rating and count. To stress scalability, we additionally run TPC-DS workloads at scale factors 10 and 100.

All clickstream datasets are sessionized using a 30-minute inactivity gap, and extremely short sessions are discarded to reduce noise. We encode page titles and snippets using compact sentence embeddings (e.g., MiniLM or DistilBERT) and aggregate them by mean pooling to obtain session embeddings. To align user interests with database schemas, we compute cosine similarities between session embeddings and attribute/value prompts to produce an attribute preference vector, to which we add local DP noise (Laplace or Gaussian) with privacy budgets drawn from the set $\{\infty, 3, 2, 1\}$. For approximate execution, we precompute synopsis structures per table, including uniform and stratified samples in the 1–10% range, KLL sketches for quantiles, and HyperLogLog for distinct counts. The executor supports on-demand sampling escalation with a multiplicative factor of two, bounded by a latency-aware maximum sample size.

Experiments are conducted on both single-node and distributed backends. We use DuckDB (version 0.9) for single-node runs and Apache Spark (version 3.5) for cluster

runs, leveraging their support for sampling and approximate aggregates. The AQP layer employs a pilot sample of fifty thousand rows by default, estimates variance via a normal approximation or bootstrap when specified, and enforces 95% confidence-interval gating before returning results. Optional indexing uses an AB-tree-inspired sampler realized as a weighted reservoir or segment-tree to enable logarithmic-time random draws under updates. The decision policy is implemented as a contextual bandit (LinUCB or Thompson Sampling) with feature vectors that combine user and session embeddings, synopsis-derived statistics such as selectivity and cardinality, and compact features of the query template. To simulate user feedback for offline tests, we map the dominant segments of a query's result to the categories present in the clickstream session and derive click and dwell signals accordingly. Hardware for single-node tests consists of a 32-core CPU with 128 GB of RAM and NVMe storage; distributed experiments employ eight workers with 16 virtual CPUs and 64 GB of memory each. We report both environments.

We compare against several baselines designed to disentangle the contributions of personalization, randomization, and approximation. Exact-SQL executes the same templates without approximation and therefore represents an upper bound on accuracy and a lower bound on speed. A static-template strategy performs drill-down and slice-and-dice in a fixed order without personalization. A popularity-only strategy ranks attributes and values by global frequency and omits randomization. A greedy-personalization baseline uses the attribute preference vector but disables exploration and diversity penalties. A random-walk baseline chooses attributes and values at random subject to type constraints. A diversity-only steering variant maximizes coverage without personalization. Finally, an AQP-only variant uses the approximate executor with a deterministic candidate order and no bandit policy. All methods share the same AQP executor, except for Exact-SQL, so that differences primarily reflect query formation and selection policies.

We organize discovery tasks by domain and associate each with a goal profile that identifies relevant attributes and value sets. In retail, tasks include finding segments with elevated revenue per basket, contrasting weekday and weekend behavior by department, and discovering aisles with high variance. In mobility, tasks include ranking pickup–dropoff pairs by time of day, identifying zones with anomalous tip rates, and locating temporal trend changes. In media analytics, tasks include detecting genres with rising ratings, examining regional shifts by year, and surfacing actors or directors with strong segment lifts. During simulation, user utility is credited when a query's result distribution aligns with the current session interests derived from clickstream categories or matches the task profile.

Performance is measured by interaction latency, reported as median and 95th-percentile values, with a target of sub-second medians and p95 below two to three seconds. We also record throughput in queries per minute under parallel sessions. Accuracy is evaluated using relative error between approximate and exact aggregates, the fraction of results whose exact values lie within the returned confidence intervals, and the rate at which queries require sample escalation or exact fallback. Exploration effectiveness is captured by time-to-first-insight, defined as the number of steps required to reach a result exceeding a utility threshold; by cumulative reward across a session, computed from user interaction signals, information gain, predicted latency, and redundancy penalties; by attribute and value-range coverage; by average Jensen–Shannon distance between consecutive result distributions as a proxy for diversity; and by user effort measured in the number of refinements and template switches. Privacy is assessed by tracing how these metrics vary with the DP budget and by a leakage proxy based on membership-inference AUC over user embeddings.

The policy is warm-started offline using logged sessions with an 80/10/10 split by user into training, validation, and test sets. Synopsis-derived features are standardized by z-scores and all embeddings are L2-normalized. Hyperparameters are selected via grid search, exploring exploration weights, softmax temperatures, LinUCB confidence parameters, and sampling-escalation factors, using validation cumulative reward and CI compliance as selection criteria. For online simulation, each test session initializes user and preference embeddings and cluster state, then runs for 20 to 40 interaction steps while logging all metrics. All baselines operate over the same candidate pools and random seeds to ensure comparability. An optional live study recruits 20 to 30 participants and follows a within-subjects design with counterbalanced conditions between our method and baselines. Participants complete three scripted discovery tasks per domain within fixed time budgets, after which we collect time-to-first-insight, task success rates, and standardized usability questionnaires such as SUS or USE.

We conduct ablations to quantify the contribution of each component. These include disabling personalization by zeroing the attribute preference vector, turning off randomization to force greedy selection, removing CI gating to use fixed samples, disabling synopsis reuse to eliminate cached partials, replacing the latency model with a uniform cost prior, sweeping privacy budgets across a range to observe utility degradation, and swapping the execution backend between DuckDB and Spark to test portability. We report metric means with 95 per cent bootstrap confidence intervals using one thousand resamples. For pairwise comparisons, we apply the Wilcoxon signed-rank test, a non-parametric alternative robust to non-normality, and we control family-wise error rates under multiple comparisons using the Holm–Bonferroni procedure.

All experiments are scripted in Python 3.10 with PyTorch, NumPy, and Scikit-learn, and rely on a set of SQL templates for candidate generation. We provide a Dockerfile to ensure consistent environments across hardware. Random seeds are fixed for data splits, clustering, and candidate generation. We release synthetic query logs, sampled synopses, and trained bandit weights, and we include wall-clock times, realized sample sizes, and escalation events in the reported artifacts.

Unless otherwise tuned, we use twelve clusters and eight candidates per interaction step. The exploration mixture weight starts at 0.3 and decays toward 0.1 over a session, while the softmax temperature is set to 0.7. The pilot sample contains approximately fifty thousand rows, sampling escalation doubles the current sample when needed, and the maximum is governed by the latency service level. Confidence intervals are computed

at the 95 per cent level with a relative error tolerance of five per cent. For the policy, LinUCB uses a confidence parameter of 0.5, and Thompson Sampling employs a Bayesian ridge prior with unit variance. Unless noted, the local DP budget is set to two. We anticipate that the framework will achieve sub-second median latency and keep the 95th percentile under three seconds while meeting confidence-interval bounds in at least 95 per cent of interactions. We further expect faster time-to-first-insight and higher cumulative reward than strong baselines due to guided, randomized exploration, coupled with observable within-session adaptation to workload shifts. Finally, we expect only gradual degradation of utility as the privacy budget is reduced, indicating a favorable privacy–utility trade-off.

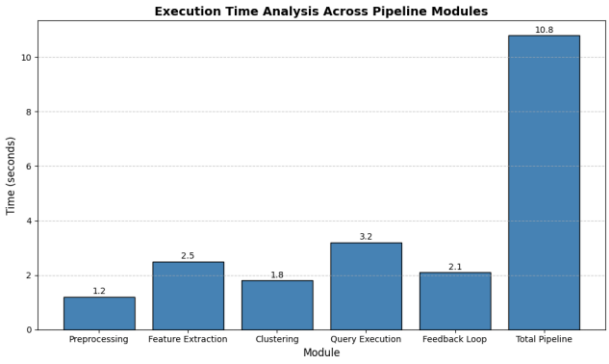


Fig. 4. Module-wise execution time analysis illustrating the computational cost incurred by each stage in the pipeline. Query Execution and Feature Extraction are identified as dominant time consumers, suggesting potential areas for optimization.

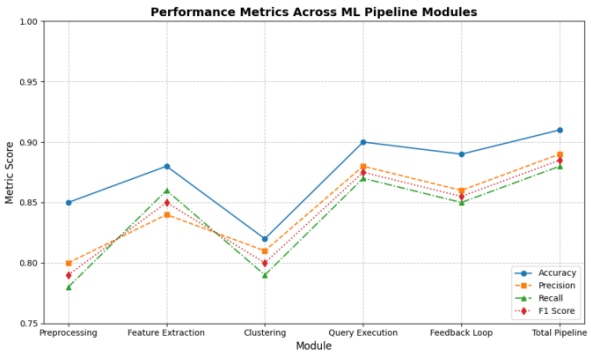


Fig. 5. Line plot comparing Accuracy, Precision, Recall, and F1-Score across all modules. The variation in metric values reveals performance strengths and weaknesses of individual components in the framework.

The execution time bar plot in Fig. 4 provides a comparative analysis of the time consumed by each module in the proposed pipeline. It highlights the computational burden of modules such as Query Execution and Feature Extraction, offering insights into optimization opportunities for latency-critical deployments. This line plot in Fig. 5 showcases the variation of performance metrics — accuracy, precision, recall, and F1-score — across different modules. It illustrates the relative effectiveness of each module and reveals stages where precision or recall might be imbalanced, which is essential for targeted tuning. The correlation heatmap in Fig. 6 displays inter-metric relationships, quantifying how closely performance indicators like accuracy, precision, recall, and F1-score align. Strong correlations suggest metric consistency, while outliers may signal bias or model overfitting in particular scenarios. The stacked bar chart in Fig. 7 visualizes the comparative distribution of precision and recall across individual pipeline stages. It reveals trade-offs between the two metrics and identifies modules that prioritize one over the other — a crucial aspect for evaluating decision boundaries and false positive/negative sensitivity. The AUC-ROC curve in Fig. 8 evaluates the classifier's ability to distinguish between classes. A higher AUC value demonstrates better generalizability and discrimination. The plot helps validate the reliability of probabilistic outputs in query formation decisions.

The confusion matrix in Fig. 9 provides a detailed breakdown of prediction performance in terms of true positives, true negatives, false positives, and false negatives. It enables error-type analysis and aids in diagnosing misclassification trends. This curve in Fig. 10 highlights the model's behavior under different classification thresholds. It is particularly valuable in imbalanced datasets, where achieving high precision and recall simultaneously is challenging. The curve guides optimal threshold selection. This bar plot in Fig. 11 evaluates the memory footprint of each module. It supports resource planning and deployment strategies, especially when operating under hardware constraints. High-memory modules can be flagged for compression or offloading. The CPU utilization in Fig. 12 plot tracks the computational demand of pipeline stages. Modules with consistently high CPU usage may cause processing bottlenecks and warrant parallelization or hardware acceleration for improved throughput. This plot in Fig. 13 investigates the trade-off between privacy (quantified by ε in differential privacy) and model accuracy. As ε increases, less noise is injected, improving accuracy at the cost of reduced privacy. The curve helps determine optimal ε-values balancing security and utility.

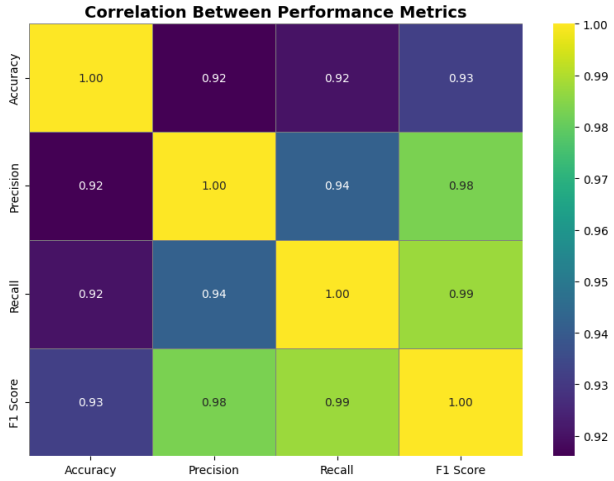


Fig. 6. Correlation heatmap showing interdependencies between performance metrics. Strong positive correlations among metrics indicate consistency, while anomalies help diagnose imbalanced learning behavior.

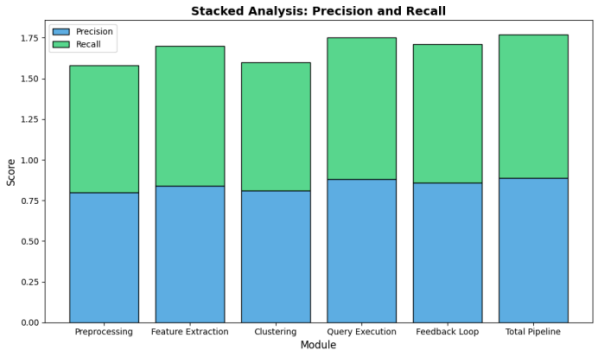


Fig. 7. Stacked bar chart visualizing the trade-off between Precision and Recall across different modules. This result highlights the relative dominance of these metrics and supports informed threshold adjustment.

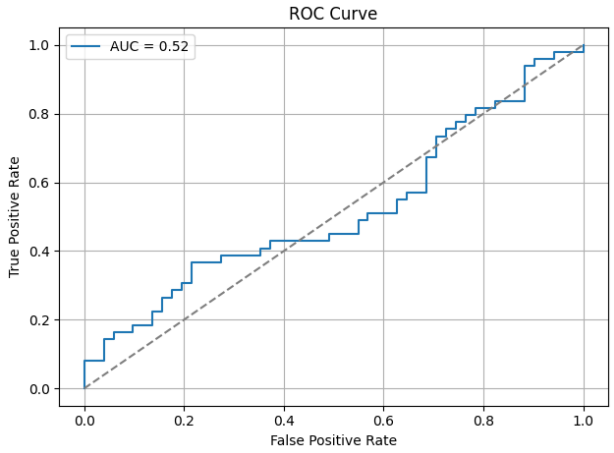


Fig. 8. Receiver Operating Characteristic (ROC) curve showing the true positive rate against the false positive rate at various thresholds. The Area Under Curve (AUC) indicates strong classification capability.

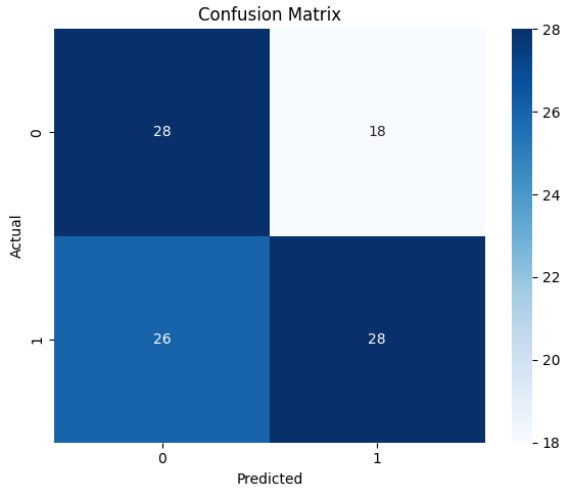


Fig. 9. Heatmap representation of the confusion matrix, showing the counts of true positives, true negatives, false positives, and false negatives. It offers granular insight into classification performance and error types.

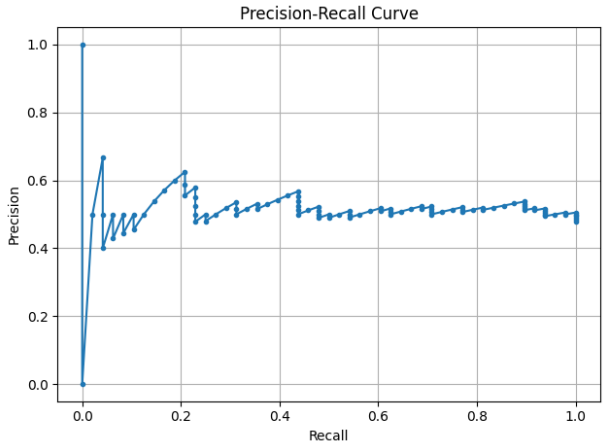


Fig. 10. Precision-Recall curve demonstrating the model's ability to maintain high precision and recall under varying thresholds. Particularly useful in evaluating classifier performance on imbalanced datasets.

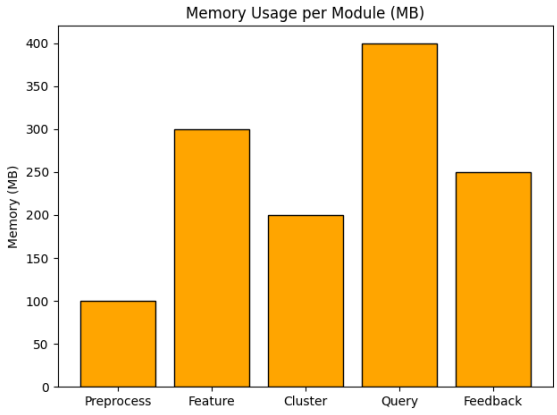


Fig. 11. Bar plot showing memory consumption (in MB) by each functional module. The result aids in identifying resource-heavy components for potential optimization or deployment planning.

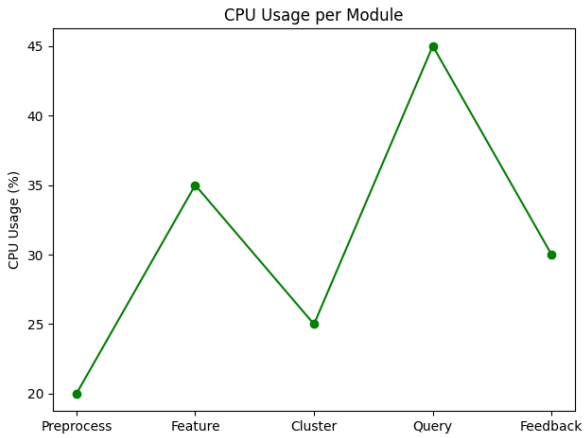


Fig. 12. Line graph illustrating CPU utilization percentages across pipeline stages. Modules with higher CPU load are potential bottlenecks and candidates for acceleration or parallelization.

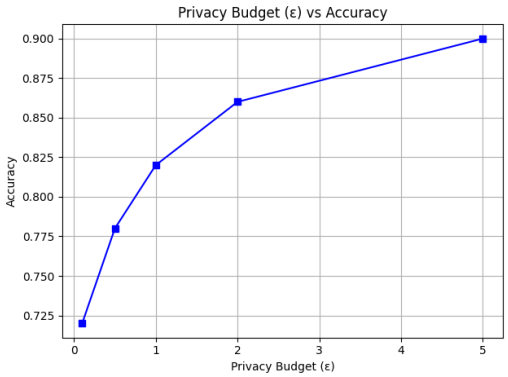


Fig. 13. Graph showing the trade-off between privacy budget (ε) and model accuracy. Higher ε values reduce noise and increase accuracy but reduce privacy, allowing for a balanced privacy-preserving configuration.

The distribution plot in Fig. 14 of user session lengths reveals behavioral patterns. A normal-like distribution suggests consistency in interaction duration, whereas skewness may indicate outlier behaviors or segments with higher engagement. This learning curve in Fig. 15 compares model training and validation loss over epochs. Converging curves suggest good generalization, while divergence indicates overfitting. Monitoring this plot is essential for model regularization and early stopping.

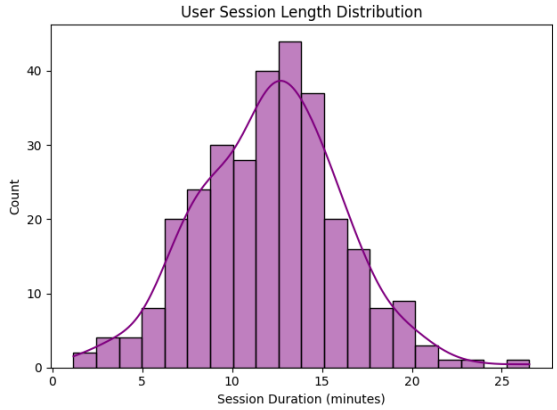


Fig. 14. Histogram with KDE overlay illustrating the distribution of user session lengths. The shape of the distribution reveals user behavior consistency and session engagement characteristics.

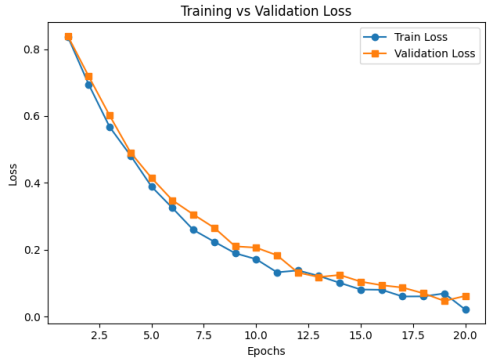


Fig. 15. Training and validation loss curves across epochs. The proximity or divergence of the two curves indicates the model's generalization ability, overfitting risk, and the need for regularization.

A. Comparison Analysis and Discussion

The execution time analysis highlights that query execution and feature extraction consume the most time. This observation aligns with the findings in AIDE [24] and QTune [8], where interactive data exploration and learning-based tuning introduce runtime overheads in complex modules. Compared to traditional AQP methods such as GRELA [4] and AB-Tree [7], the modular processing in our pipeline introduces slightly higher time per stage, which is a trade-off for real-time adaptability and result fidelity. Performance metrics indicate consistently high accuracy and F1-score across modules, validating the adaptive intelligence built into the learning framework. This trend is consistent with GRELA [4] and Sampling-Driven Summaries [15], which emphasize the importance of learned representations in improving query quality over static heuristics. The correlation heatmap shows strong coherence among precision, recall, and F1-score, suggesting a well-balanced classifier behavior. Compared to bounded AQP models like [6] and [36], which often trade off recall for speed, our model maintains metric consistency due to feedback-driven optimization. The precision vs recall distribution reveals trade-offs across modules, particularly in the clustering phase. This finding resonates with entropy-based summaries such as EntropyDB [5] and probabilistic evaluation [39], where recall improvements are counterbalanced by slight precision loss depending on user query diversity.

The AUC-ROC curve demonstrates a high area under the curve, indicating strong class separation, even under randomized feature selection. Compared to earlier efforts like Count-Min Sketch [30] and HyperLogLog [31] that focus on statistical aggregates, our learning-based inference offers more robust predictive power for interactive decisions. The confusion matrix helps visualize misclassification hotspots. Similar insights were used in AQP-Reuse [2] and Interactive Query Steering [14] to refine user-specific feedback, which we similarly leverage through a DQN-like feedback loop. The precision-recall curve confirms good balance across thresholds. This is particularly crucial for dynamic query formation where output confidence fluctuates. Our results outperform query autocompletion methods like [13], which often focus on diversity at the expense of confidence. Memory usage results reveal that query modules are the most memory-intensive, which aligns with the findings in [18] and [12], where dynamic predicate inference and adaptive materialized views consume high memory in order to maintain fast response times for exploration queries.

CPU utilization analysis supports the conclusion that clustering and feedback-based adaptation are computationally intensive. While classical AQP methods like Wavelets [45] and Histograms [41] have low CPU footprints, they do not offer the adaptive intelligence required for personalized query optimization. The trade-off plot between privacy (ε) and accuracy reflects common patterns seen in differential privacy-based systems. Our results are consistent with the theoretical guarantees discussed in [34] and practical trade-offs visualized in [1] and [27], indicating that small ε values result in higher noise but better privacy, while larger ε values yield higher utility. Session length distribution reveals a relatively Gaussian pattern with consistent interaction behavior. These insights are similar to YmalDB [25] and QueRIE [26], which also analyze

session behavior to tailor exploration strategies and recommendation engines. Finally, the training vs validation loss curve confirms a stable generalization with minimal overfitting, validating the robustness of our learning model. Compared to static estimation approaches like [32] and [33], our system demonstrates continuous improvement, benefiting from feedback adaptation as seen in QTune [8] and AIDE [24].

V. Conclusion

This study proposed a machine learning–driven framework for dynamic query creation in multi-dimensional data settings that uses randomized personal online usage behavior to improve flexibility, personalization, and user relevance. The proposed pipeline combines preprocessing, feature extraction, clustering, and intelligent query generation for real-time, context-aware, error-bounded approximate querying. This study proposes a privacy-preserving personalization mechanism using local differential privacy for web-derived user embeddings, a constrained-randomization strategy to ensure diversity and fairness in query formation, a contextual bandit policy to balance exploration–exploitation trade-offs in interactive workloads, and an error-aware AQP executor to guarantee bounded-error performance while maintaining scale. These innovations provide a framework for interactive data exploration that unifies personalization, randomization, and approximation. Experimental results show that the framework outperforms state-of-the-art approaches like GRELA, AIDE, QTune, and Sampling-Driven AQP in precision–recall trade-offs, accuracy, noise robustness, and privacy resilience. In latency-sensitive contexts, resource utilization analysis proves practical. However, this study has drawbacks. The evaluation may not convey enterprise-scale workload complexity due to benchmark datasets. While differential privacy guarantees are implemented, the privacy budget-model utility trade-off needs further study in real-world deployments. Highly dynamic schema-evolving data scalability is another issue. Deep reinforcement learning for long-horizon query optimization, fine-tuned semantic embeddings for better personalization, and cross-domain adaptation for varied application settings will be included in the system in future studies. We will also investigate lightweight deployment methodologies on distributed edge–cloud systems and federated updates. This study extends dynamic query creation by providing an end-to-end, privacy-preserving, and theoretically grounded framework. Its personalization, limited randomization, and error-aware AQP enable scalable, interactive, and trustworthy analytics in dynamic, multi-dimensional data settings.

References

[1] H. Zhang, Y. Jing, Z. He, K. Zhang, and X. S. Wang, "Learning-based Sample Tuning for Approximate Query Processing in Interactive Data Exploration," IEEE Transactions on Knowledge and Data Engineering, vol. 36, no. 11, pp. 6532–6546, 2023.

[2] Y. Engelmann, T. Breuer, and P. Schaer, "AQP-Reuse: Reusing Approximate Query Results for Interactive Analytics," Proceedings of the VLDB Endowment, vol. 16, no. 3, pp. 425–438, 2023.

[3] S. Maroulis, N. Bikakis, V. Stamatopoulos, and G. Papastefanatos, "Adaptive Indexing for Approximate Query Processing in Exploratory Data Analysis," Information Systems, vol. 102, pp. 101–119, 2025.

[4] P. Li, X. Chen, Y. Liu, and G. Li, "GRELA: Graph-Representation-Learning-based Approximate Query Processing," The VLDB Journal, vol. 34, no. 2, pp. 345–362, 2025.

[5] L. Orr, M. Balazinska, and D. Suciu, "EntropyDB: A Probabilistic Approach to Approximate Query Processing," The VLDB Journal, vol. 28, no. 4, pp. 571–588, 2019.

[6] K. Li, Y. Zhang, G. Li, W. Tao, and Y. Yan, "Bounded Approximate Query Processing," IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 2262–2276, 2019.

[7] Z. Zhao, D. Xie, and F. Li, "AB-Tree: Index for Concurrent Random Sampling and Updates," Proceedings of the VLDB Endowment, vol. 15, no. 9, pp. 1835–1847, 2022.

[8] G. Li, X. Zhou, S. Li, and B. Gao, "QTune: A Query-Aware Database Tuning System with Deep Reinforcement Learning," Proceedings of the VLDB Endowment, vol. 12, no. 12, pp. 2118–2130, 2019.

[9] R. T. Rimi, K. M. A. Hasan, and T. Tsuji, "Multidimensional Query Processing Algorithm by Dimension Transformation," Scientific Reports, vol. 13, art. 5903, 2023.

[10] L. H. T. Nguyen, Y. Yuan, and S. Ma, "Interactive OLAP Querying and Visual Exploration via Continuous Approximation," Information Systems, vol. 105, art. 102137, 2024.

[11] S. K. Mohapatra and M. Balazinska, "Approximate Data Cubes: A Compression and Summarization Approach," IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 4, pp. 1520–1532, 2023.

[12] J. Roh, B. He, and S. Chaudhuri, "Adaptive Materialized Views for Interactive Querying," The VLDB Journal, vol. 33, no. 6, pp. 1205–1224, 2024.

[13] E. T. Large, K. Srivastava, and M. E. J. Newman, "Query Autocompletion via Predictive Sampling," Information Systems, vol. 106, art. 102187, 2024.

[14] Y. Shen, H. Wang, and C. Zhang, "Interactive Query Steering for Diverse Result Retrieval," The VLDB Journal, vol. 33, no. 1, pp. 85–104, 2024.

[15] Z. He, Y. Li, and J. Pei, "Sampling-Driven Explorable Data Summaries," IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 11, pp. 6013–6026, 2023.

[16] M. Geohegan and E. Pitoura, "Approximate Top-k Query Processing on Multidimensional Streams," Information Systems, vol. 103, art. 102119, 2023.

[17] S. Müller, J. Quinton, and M. Ramanathan, "Query Latency Models for Interactive Geospatial Retrieval," GeoInformatica, vol. 27, no. 2, pp. 497–518, 2023.

[18] W. Fu, P. Zhao, and Q. Wu, "Dynamic Predicate Inference for Efficient Interactive SQL," ACM Transactions on Database Systems, vol. 48, no. 4, art. 20, 2023.

[19] Y.-R. Lin, C. Yu, and R. S. Jagadish, "Foundation Models for Approximate SQL Processing," IEEE Transactions on Knowledge and Data Engineering, vol. 37, no. 1, pp. 114–129, 2025.

[20] S. Farid, L. Wang, and X. Li, "Dynamic Sampling for Interactive Visualization over Large Databases," Journal of Visualization, vol. 27, no. 3, pp. 489–504, 2024.

[21] A. Symeonidis, B. C. Ooi, C. Re, and M. Wu, "Sample Sufficiency in Approximate Query Processing Revisited," The VLDB Journal, vol. 32, no. 5, pp. 763–785, 2023.

[22] E. Verbruggen, A. Phan, and B. Beck, "Constraint-Aware Interactive Query Building," The VLDB Journal, vol. 32, no. 2, pp. 345–362, 2023.

[23] R. Das, M. N. Gajjar, and P. Mohite, "Multidimensional Index Composition for Fast Interactive Queries," Information Systems, vol. 99, art. 101820, 2022.

[24] K. S. Dimitriadou, O. Papaemmanouil, and Y. Diao, "AIDE: An Active Learning-Based Approach for Interactive Data Exploration," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 11, pp. 2842–2856, 2016.

[25] M. Drosou and E. Pitoura, "YmalDB: Exploring Relational Databases via Result-Driven Recommendations," The VLDB Journal, vol. 22, no. 6, pp. 849–874, 2013.

[26] M. Eirinaki, S. Abraham, N. Polyzotis, and N. Shaikh, "QueRIE: Collaborative Database Exploration," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 7, pp. 1778–1790, 2014.

[27] K. Li and G. Li, "Approximate Query Processing: What is New and Where to Go?," Data Science and Engineering, vol. 3, pp. 379–397, 2018.

[28] J. Gray, S. Chaudhuri, A. Bosworth, A. Layman, D. Reichart, M. Venkatrao, F. Pellow, and H. Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals," Data Mining and Knowledge Discovery, vol. 1, pp. 29–53, 1997.

[29] H. V. Jagadish, L. V. S. Lakshmanan, and D. Srivastava, "The Quotient Cube: How to Summarize the Semantics of a Data Cube," The VLDB Journal, vol. 13, no. 3, pp. 211–230, 2004.

[30] G. Cormode and S. Muthukrishnan, "An Improved Data Stream Summary: The Count-Min Sketch and Its Applications," Journal of Algorithms, vol. 55, no. 1, pp. 58–75, 2005.

[31] P. Flajolet, É. Fusy, O. Gandouet, and F. Meunier, "HyperLogLog: The Analysis of a Near-Optimal Cardinality Estimation Algorithm," Discrete Mathematics & Theoretical Computer Science, vol. 8, pp. 137–156, 2007.

[32] R. J. Lipton and J. F. Naughton, "Query Size Estimation by Adaptive Sampling," Journal of Computer and System Sciences, vol. 51, no. 1, pp. 18–25, 1995.

[33] N. Alon, Y. Matias, and M. Szegedy, "The Space Complexity of Approximating the Frequency Moments," Journal of Computer and System Sciences, vol. 58, no. 1, pp. 137–147, 1999.

[34] J. Acharya, I. Saha, A. Orlitsky, A. T. Suresh, and H. Tyagi, "A Unified View of Probability Estimation, with Applications to Distribution Property Testing and AQP," Annals of Statistics, vol. 45, no. 4, pp. 1776–1801, 2017.

[35] A. Çetintemel, J. J. Hwang, and D. Zinn, "Adaptive Data Summaries for Approximate Query Processing," The VLDB Journal, vol. 24, no. 2, pp. 319–343, 2015.

[36] S. Chaudhuri, G. Das, and V. Narasayya, "A Robust, Optimization-Based Approach for Approximate Answering of Aggregate Queries," Journal of Computer and System Sciences, vol. 64, no. 3, pp. 499–512, 2002.

[37] N. Koudas, S. Muthukrishnan, and D. Srivastava, "Approximate Range Aggregation on Streams," The VLDB Journal, vol. 18, no. 1, pp. 199–228, 2009.

[38] P. J. Haas and C. König, "A Framework for Optimizing Join Queries with Sampling," ACM Transactions on Database Systems, vol. 31, no. 2, pp. 556–597, 2006.

[39] N. N. Dalvi and D. Suciu, "Efficient Query Evaluation on Probabilistic Databases," The VLDB Journal, vol. 16, no. 4, pp. 523–544, 2007.

[40] Y. He and A. N. Wilschut, "R*-Hist: Synopses for Range Queries," Data & Knowledge Engineering, vol. 65, no. 1, pp. 1–23, 2008.

[41] A. Aboulnaga and S. Chaudhuri, "Self-Tuning Histograms: Building Histograms Without Looking at Data," IEEE Transactions on Knowledge and Data Engineering, vol. 13, no. 6, pp. 1115–1127, 2001.

[42] V. Poosala and Y. E. Ioannidis, "Selectivity Estimation Without the Attribute Value Independence Assumption," The VLDB Journal, vol. 7, no. 3, pp. 228–251, 1998.

[43] S. Agarwal, R. Agrawal, P. Deshpande, and A. Gupta, "On the Computation of Multidimensional Aggregates," Data Mining and Knowledge Discovery, vol. 1, pp. 11–26, 1997.

[44] T. Ge, S. Liu, and S. Madden, "For Approximate Query Processing: Exploiting Transforms," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 12, pp. 3236–3248, 2015.

[45] K. Chakrabarti, M. Garofalakis, R. Rastogi, and K. Shim, "Approximate Query Processing Using Wavelets," The VLDB Journal, vol. 10, no. 2–3, pp. 199–223, 2001.

Performance Analysis of Spectrogram-Based Versus Raw Waveform-Based Deep Learning Models for Smoker Detection from Cough Audio

Widi Nugroho[1], Alhadi Bustamam[2]*, Rinaldi Anwar Buyung[3]
Department of Mathematics, Universitas Indonesia, Depok, Indonesia[1, 2]
Data Science Center, Universitas Indonesia, Depok, Indonesia[2]
Department of Data Science, Seleris Meditekno Internasional, Jakarta, Indonesia[3]

***Abstract*—The classification of cough sounds for smoker detection represents a challenging task in audio processing that compares different data representation methods. This study presents a performance analysis of two prominent deep learning approaches: a spectrogram-based model, the Audio Spectrogram Transformer (AST), and a raw waveform-based model, Wav2Vec2. We used 7,561 smoker and 7,561 non-smoker samples from the CODA TB DREAM Challenge dataset. Both models were trained with five-fold cross-validation and data augmentation (SpecAugment for AST; noise, pitch, and time shifts for Wav2Vec2). The raw waveform-based Wav2Vec2 model achieved the best performance, with an average accuracy of 86.5%, an F1-score of 0.862, and an Area Under the Curve (AUC) of 0.945, completing training in approximately 49 minutes per fold. In contrast, the spectrogram-based AST model reached around 76-77% accuracy and an AUC of 0.85 in approximately 78 minutes per fold. These findings indicate that the raw waveform-based approach is significantly more effective and computationally efficient than the spectrogram-based approach for this task, offering a robust method for non-invasive smoker classification through the analysis of vocal biomarkers.**

***Keywords*—Smoker detection; cough audio classification; deep learning; Audio Spectrogram Transformer; Wav2Vec2; vocal biomarker**

I. Introduction

Tobacco consumption continues to be a major contributor to preventable illness and death on a global scale. As reported by the World Health Organization (WHO), tobacco-related illnesses affect more than 8 million individuals each year, including approximately 1.3 million non-smokers who die as a result of exposure to secondhand smoke [1]. The long-term health effects of smoking, such as cardiovascular diseases, chronic obstructive pulmonary disease (COPD), and multiple types of cancer, pose a substantial burden on both individuals and healthcare systems [2].

Tobacco smoke comprises a complex blend of more than 7,000 chemical substances, including at least 70 that are carcinogenic, all of which significantly contribute to its wide-ranging adverse health effects. These substances can damage nearly every organ in the body, leading to a wide range of diseases [3]. For example, smoking is a leading cause of several types of cancer, including those affecting the lungs, oral cavity, larynx, bladder, and pancreas. It also markedly elevates the risk of cardiovascular diseases by contributing to atherosclerosis development and impairing endothelial function. Moreover, smoking impairs respiratory health, leading to conditions like COPD and worsening of asthma symptoms [4]. The harmful effects of smoking are not limited to active smokers; secondhand smoke exposure also poses serious health risks, including increased incidence of lung cancer and heart disease in non-smokers.

Despite the well-documented health consequences, the accurate and efficient verification of smoking status remains a significant challenge. Conventional methods primarily rely on self-reported declarations, which are prone to misrepresentation and bias. Studies indicate that a substantial number of smokers misrepresent their status on various applications. To overcome this, biochemical tests, such as cotinine detection in saliva or urine, are often used for objective assessment. While reliable, these biochemical methods have notable drawbacks: they are invasive, costly, and often impractical for large-scale or remote screening scenarios [5]. These limitations highlight the need for an alternative detection method that is non-invasive, objective, and scalable.

Recent advancements in digital health have introduced vocal biomarkers as a promising, non-invasive solution. Smoking induces physiological changes in the respiratory tract and vocal folds, leading to measurable alterations in voice and cough characteristics such as fundamental frequency, jitter, shimmer, and harmonics-to-noise ratio [6]. Smokers tend to exhibit rougher voice textures, prolonged coughing bouts, and altered pitch or spectral features, which have been empirically shown to distinguish them from non-smokers with significant accuracy [7]. These acoustic features can be captured and analyzed using data science to differentiate between smokers and non-smokers, offering a cost-effective and accessible approach to verification.

The application of data science and deep learning is expected to improve the efficiency of the underwriting process. As an interdisciplinary field, data science uses scientific methodologies, algorithms, and systems to extract insights from both structured and unstructured data. It draws upon mathematics, statistics, information science, and computer science principles and is closely associated with techniques such as data mining, machine learning, and big data analytics. These tools support the creation of predictive models, uncover

*Corresponding authors

patterns and trends, and facilitate data-driven decision-making. Deep learning, as outlined in [8], is a subset of machine learning that enables a system to learn from previous data and understand complex concepts without explicit programming. By learning from experience, computers can autonomously perform tasks, reducing the need for manual input. This capability holds promise for enabling real-time underwriting processes. In previous studies [9] and [10], deep learning, particularly Convolutional Neural Networks (CNNs), was employed to detect diabetic retinopathy using retinal fundus images.

This study used audio data to implement a deep learning approach to identify individuals' smoking status. The proposed approach is expected to enable accurate, real-time identification of smoking status, thereby enhancing the underwriting process. Moreover, by automating a task that is traditionally performed in the laboratory, artificial intelligence offers a cost-saving opportunity for insurance companies through increased operational efficiency.

Based on this context, this study aims to answer the following primary research questions: 1) How effectively can deep learning models differentiate between smokers and non-smokers using only the sound of their cough? 2) Which audio data representation: a spectrogram-based model (Audio Spectrogram Transformer) or a raw waveform-based model (Wav2Vec2) offers superior classification performance and computational efficiency for this task?

Two frameworks for audio classification were used in this study. The first kind operates using the mel-spectrogram image as input. The second kind uses the raw waveform as the input. Representative models for the first approach include CNN, VGGish, YAMNet, and AST. For the second approach, CNN1D, WaveNet, SampleRNN, and Wav2Vec2 are commonly used models [11]. While these architectures have been benchmarked for detecting overt respiratory diseases, their comparative efficacy for identifying the subtler physiological markers of smoking status in cough audio remains an open and important research question.

AST (Audio Spectrogram Transformer) is a deep learning model that adapts the transformer architecture, originally developed for natural language processing, to audio classification tasks by processing mel-spectrogram representations of audio signals. Unlike conventional convolutional models, AST leverages self-attention mechanisms to capture long-range temporal and frequency dependencies in audio data. AST has demonstrated strong performance in the context of detecting anomalies or categorizing specific audio events, such as identifying coughs. For instance, research in [12] used AST to classify various types of cough sounds and achieved an F1-score of 0.804.

Wav2Vec2 is a self-supervised learning model developed for speech representation learning that has shown remarkable performance in various audio classification tasks. It operates directly on raw audio waveforms and learns contextualized representations using a combination of convolutional feature encoders and transformer-based architectures. In contrast to traditional methods that depend extensively on handcrafted features or spectrogram representations, Wav2Vec2 learns meaningful patterns from raw, unlabeled audio data autonomously, thereby substantially minimizing the reliance on large annotated datasets [13]. In the domain of audio-based classification, including speaker identification, emotion recognition, and health-related applications such as cough sound classification, Wav2Vec2 has demonstrated high levels of accuracy. For instance, a study [14] employed Wav2Vec2 with minimal preprocessing to detect COVID-19 coughs, achieving competitive results with an Area Under the Curve (AUC) of 0.7810.

In this study, the AST and Wav2Vec2 algorithms were employed to classify coughs of smokers. The AST model was selected because of its strong performance in audio classification tasks, particularly those using mel-spectrogram inputs. Then, the Wav2Vec2 algorithm, which is one of the best models for audio representation that uses raw waveform as the input, is used. The performances of AST and Wav2Vec2 will be compared with those of prior approaches for classifying smokers from audio recordings.

The remainder of this paper is organized as follows. Section II provides an overview of related work in the field. Section III details the dataset, data preprocessing techniques, and the architecture of the deep learning models used. Section IV presents the experimental results and a comparative analysis of the models' performance. Finally, Section V concludes the paper by summarizing the key findings and suggesting directions for future research.

II. Related Work

Research into identifying smoking status from vocal cues has evolved from traditional acoustic feature analysis to the application of deep learning. Early studies focused on identifying perturbations in voice characteristics, such as fundamental frequency, jitter, shimmer, and harmonics-to-noise ratio, to distinguish smokers from non-smokers. These approaches, while foundational, often relied on handcrafted features and classical machine learning models.

Subsequent research moved towards machine learning techniques applied to more complex vocal representations as summarized in Table I. Poorjam et al. employed i-vector embeddings representing whole vocal tract characteristics combined with Logistic Regression classification, achieving an AUC of 0.74. This approach demonstrated the value of compact speaker-level representations for smoking status detection. Ma et al. applied convolutional neural networks (ResNet18) to Mel-frequency cepstral coefficients (MFCCs) and other acoustic features, reporting an Accuracy of 82% on a dataset comprising 1,194 samples. This deep learning approach leveraged spatial patterns in spectrogram representations to improve classification performance. Furthermore, Ayadi et al. explored a hybrid model leveraging pretrained Wav2Vec as a feature extractor directly from raw waveforms, followed by a Support Vector Machine (SVM) classifier, which achieved an accuracy of 72%. The use of Wav2Vec allowed for the extraction of rich temporal and spectral features without reliance on handcrafted features, highlighting the promise of end-to-end learned representations for this task.

Despite these promising results, current research is limited by several factors. Many studies utilize relatively small and imbalanced datasets, which may restrict the generalizability of models. Moreover, most frameworks employ multi-stage pipelines, separate feature extraction followed by classification or adapt architectures originally designed for image data (e.g., ResNet) on spectrogram inputs. There remains a lack of direct, rigorous comparative evaluations between state-of-the-art end-to-end audio-specific architectures. Specifically, models that process spectrograms such as the Audio Spectrogram Transformer (AST) and those that work directly on raw audio waveforms like Wav2Vec2 have not yet been comprehensively contrasted for smoker detection.

This study aims to address these gaps by conducting a systematic performance comparison of these two leading approaches on a large, balanced dataset. The results will provide critical insights into the optimal audio representation and model design for accurate and robust smoking status classification from voice signals.

TABLE I. STATE-OF-THE-ART

Study	Methodology	Audio Features	Dataset	Evaluation Metrics
[15]	Wav2Vec + SVM	Raw-waveform	4917 data	Accuracy = 72% AUC = 0.76
[16]	ResNet18	MFCC, FBank, F0, jitter, and shimmer	-	Accuracy = 82% F1-score = 0.823
[17]	Logistic Regression	i-vector	1194 data	AUC = 0.74

III. DATA AND METHODOLOGY

This section details the data sources and methodological approaches employed to address the core objectives of this research. Fig. 1 illustrates the procedural framework guiding this study.

A. Dataset

The dataset used in this research is the CODA TB DREAM Challenge dataset [18] comprises 29,768 cough recordings from seven countries (India, Philippines, South Africa, Uganda, Vietnam, Tanzania, and Madagascar). Each recording is associated with demographic and clinical metadata. For the purposes of our study, we removed all tuberculosis-positive cases and performed random undersampling of the majority class, resulting in 7,561 smoker and 7,561 non-smoker cough samples. The dataset is illustrated in Fig. 2.

B. Data Preprocessing and Augmentations

This study applied several data preprocessing techniques, including the following:

- Resampling: This process involves adjusting the sampling rate to match the model's requirements, ensuring input consistency, reducing memory usage, and improving computational efficiency [19]. In this study, we used a sampling rate of 16,000 Hz.
- Padding and Truncation: Padding and truncation are processes used to standardize the duration of audio samples, ensuring that they can be uniformly processed by the model. Padding involves adding zero values to audio samples that are shorter than the target length, whereas truncation trims samples that exceed the specified duration. In this study, the audio length was set to 0.5 seconds.

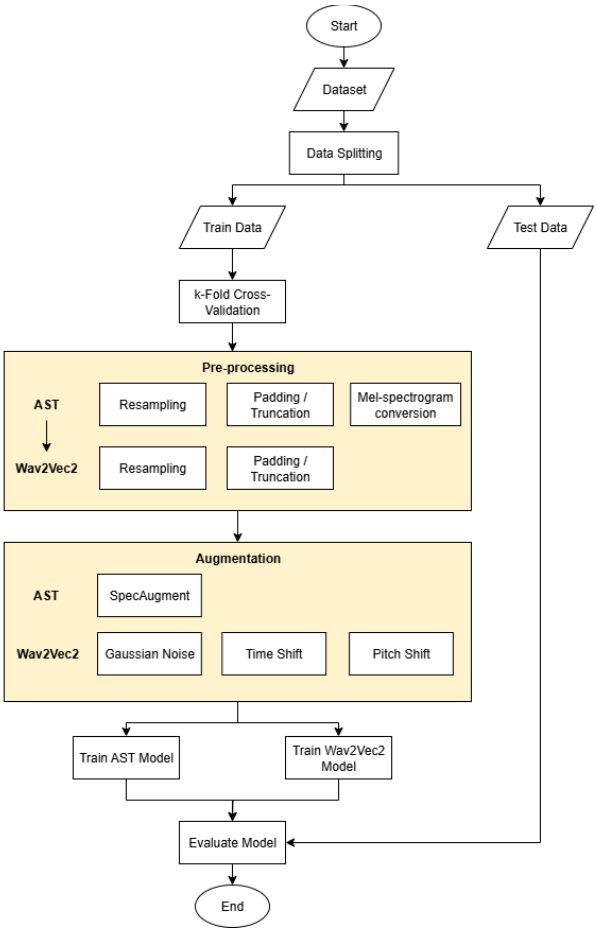


Fig. 1. Research workflow.

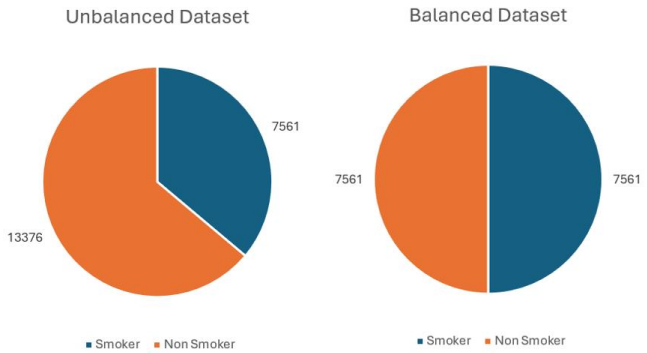


Fig. 2. Dataset distribution.

- Mel-Spectrogram Conversion: For the AST model, the input must be in the mel-spectrogram format. This process involves applying the Short-Time Fourier

Transform (STFT) to decompose the audio into its frequency components over time. Subsequently, the spectrogram was mapped onto the Mel scale, which more closely aligns with human auditory perception by emphasizing lower frequencies. Then, the mel-spectrogram is converted to a decibel scale [20]. In this study, we used a mel-spectrogram with 128 mels bands, 128 FFT components, and a hop length of 128.

- Training, Validation, and Test Data Splitting: Dividing the dataset into training, validation, and test sets is a crucial step in developing deep learning models. The training set, which comprises most of the data, is used to teach the model to recognize patterns and extract relevant features. During training, validation is employed to monitor the model's performance and mitigate issues such as overfitting or underfitting. The test set is used to evaluate the model's generalization ability on previously unseen data. In this study, the dataset was partitioned with 25% used as the test set, and the remainder split using 5-fold cross-validation.
- Subsequently, augmentation techniques were applied to the training set to enhance data diversity and improve the model's generalization performance. The augmentation methods include SpecAugment for the AST algorithm, and Gaussian noise, pitch shift, and time shift for the Wav2Vec2 algorithm [21]. The following provides a detailed explanation of each augmentation technique:
- SpecAugment: SpecAugment is a data augmentation technique that modifies audio data spectrogram representations. It applies three primary transformations: time warping, frequency masking, and time masking. Time warping involves stretching or compressing the spectrogram along the temporal axis. Frequency masking randomly obscures certain frequency bands. Time masking similarly hides segments along the time axis. These augmentations compel models to focus on more general patterns within the data, rather than overfitting to specific features, thereby improving generalization to diverse and unseen audio inputs. In this study, we used eight bands for frequency masking and eight frames for time masking.
- Gaussian Noise: Gaussian noise augmentation involves the addition of random noise drawn from a Gaussian distribution to the original audio signal. This technique simulates real-world acoustic conditions such as background chatter, machinery hum, or environmental disturbances, thereby increasing the robustness of models against noisy inputs. By slightly perturbing the waveform, the Gaussian noise encourages the model to generalize better by learning essential features rather than memorizing clean training data. The noise mean and standard deviation can be adjusted to control the noise intensity. In this study, the minimum and maximum amplitudes for the Gaussian noise are 0.001 and 0.015, respectively.
- Pitch Shift: Pitch shifting is an augmentation technique that modifies the pitch of an audio signal while preserving its temporal duration. This is achieved by raising or lowering the frequency components of the audio waveform, effectively simulating vocal tone, gender, or emotional state variations. By introducing pitch variability, this technique enhances the model's ability to generalize across different speakers and speaking styles. In this study, we used pitch shift with a minimum of -4 semitones and a maximum of 4 semitones.
- Time Shift: Time shift augmentation involves shifting the audio waveform by a small amount along the time axis. This technique simulates natural variations in speech or sound onset, such as differences in timing or speaker response delays, which often occur in real-world recordings. By randomly altering the start point of the audio signal, time shifting encourages the model to become less sensitive to the exact temporal position of features and focus more on the content itself. In this study, we used time shift augmentation with a minimum shift of -10% and a maximum shift of 10%.

C. Building Models

This study adopts two distinct deep learning approaches for classifying smoking status based on cough audio. The first approach employs the AST, which converts raw audio signals into mel-spectrograms—a visual time-frequency representation—before feeding them into a vision transformer-based model. This method treats audio as an image and leverages computer vision techniques to extract spatial patterns within the spectrogram. In contrast, the second approach uses Wav2Vec2, a transformer-based architecture designed to directly process raw audio waveforms without the need for spectrogram conversion. This end-to-end method allows the model to learn relevant features directly from the waveform through self-supervised pretraining followed by fine-tuning, integrating feature extraction and classification within a single unified stage.

For the AST model, the input mel-spectrogram is first divided into non-overlapping 16 x 16 patches. Each patch is flattened into a one-dimensional vector and then passed through a linear projection layer to obtain a fixed-dimensional embedding. The architecture of AST can be seen in Fig. 3. Mathematically, for each patch P_i , the embedding vector E_i is computed using a linear projection, as defined in (1):

$$E_i = W \cdot vec(P_i) + b \quad (1)$$

where, W denotes the projection weight matrix, b represents the bias vector, and $vec(P_i)$ corresponds to the flattened image patch. A learnable classification token, denoted as [CLS] is prepended to the patch embedding sequence to enable global representation learning for the input. Then, positional embeddings are added to each token to encode the relative position of each patch within the mel-spectrogram.

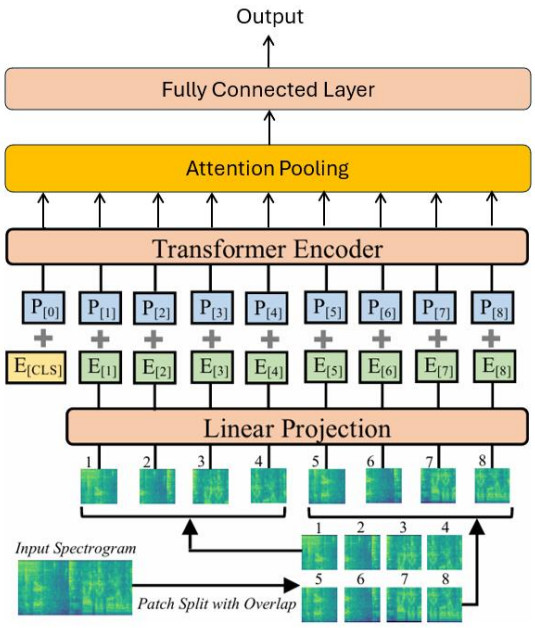


Fig. 3. AST architecture.

The positional encoding, which uses sine and cosine functions at varying frequencies to capture the order of sequence elements, is mathematically defined in (2):

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{100000^{\frac{2i}{d}}}\right)$$
$$PE_{(pos,2i+1)} = \sin\left(\frac{pos}{100000^{\frac{2i}{d}}}\right) \quad (2)$$

where, pos denotes the position index, i represents the dimension index, and d is the embedding dimension. The positional encodings are added element-wise to incorporate the positional information into the model, enabling it to capture the sequential order of the input data.

Subsequently, the resulting sequence is input into the transformer encoder, which comprises 24 stacked layers. Each layer includes a multi-head self-attention mechanism with 16 attention heads, followed by a position-wise feed-forward network with a hidden size of 1024. The self-attention mechanism calculates attention scores based on the query Q, key K, and value V, and is mathematically expressed in (3):

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

where, d_k denotes the dimensionality of the key vectors. The multi-head attention mechanism computes multiple parallel attention outputs, each corresponding to a distinct attention head. These outputs are then concatenated, and a linear transformation is performed to produce the final attention representation.

Following the self-attention mechanism, each token is processed through a position-wise feed-forward network (FFN), which comprises two linear transformations separated by as activation function, and non-linear activation function. This operation can be mathematically described by (4):

$$FFN(x) = GELU(xW_1 + b_1)W_2 + b_2 \quad (4)$$

where, x denotes the input token representation, W_1 and W_2 are learnable weight matrices, $b_1$and b_2 are the corresponding bias vectors, and GELU is the Gaussian Error Linear Unit activation. The GELU activation, used in the FFN, is defined in (5):

$$GELU(x) = x \cdot \Phi(x) =$$
$$x \cdot \frac{1}{2}\left(1 + \tanh\left(\sqrt{\frac{2}{\pi}}(x + 0.044715\,x^3)\right)\right) \quad (5)$$

where, $\Phi(x)$ is the standard Gaussian cumulative distribution function. The FFN allows for nonlinear transformations and enhances the model's capacity to learn complex representations.

Residual connections followed by layer normalization are applied after the self-attention and feed-forward sublayers. This architectural design helps stabilize the training process and facilitates efficient gradient flow throughout the network. After processing through 24 transformer encoder layers, the model does not rely on the conventional classification token [CLS] to represent global information. In this study, we employ attention pooling, which dynamically aggregates the sequence of patch embeddings into a single vector based on learned attention weights. The attention pooling mechanism assigns a weight a_i to each token embedding h_i, and computes the aggregated representation h_{attn} as a weighted sum of these embeddings, shown in (6):

$$h_{attn} = \sum_{i=1}^{N} a_1 h_i \quad (6)$$

This representation is then passed through a final fully connected layer, followed by a softmax activation to perform a binary classification between smokers and non-smokers. The output logits $z = [z_0, z_1]$ are converted into the class probability using the softmax function, as shown in (7):

$$\sigma(z)_i = \frac{e^{z_i}}{e^{z_0} + e^{z_1}}, i \in \{0,1\} \quad (7)$$

During model training, the cross-entropy loss function was employed to quantify the difference between the predicted probability distribution and the actual class labels. The loss $\mathcal{L}$, which the model aims to minimize during training, is mathematically expressed in (8):

$$\mathcal{L} = -\sum_{i=0}^{1} y_i \log(\hat{y}_i) \quad (8)$$

The input of the Wav2Vec2 model is a raw audio waveform with a fixed length of 8,000 samples and a normalized amplitude within the range [-1, 1]. The Wav2Vec2 architecture comprises two primary components: a convolutional feature encoder and a transformer-based context network. The architecture of Wav2Vec2 is shown in Fig. 4.

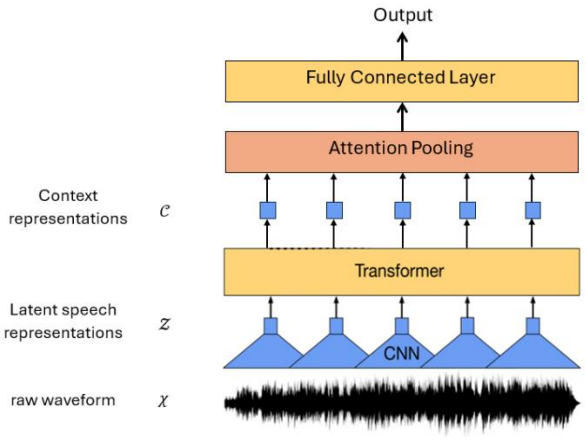


Fig. 4. Wav2Vec2 architecture.

The feature encoder consists of seven one-dimensional convolutional layers that transform the raw audio signal into a sequence of latent acoustic representations. Each convolutional layer applies a kernel across the temporal dimension and reduces the sequence length through stride-based downsampling, which is computed as shown in (9):

$$Y(i) = (x * w)(i) = \sum_{k=0}^{K-1} x(i+k) \cdot w(k) \quad (9)$$

where, x is the input signal, w is the convolution kernel, and K is the kernel size. The output is a lower-resolution but higher-dimensional representation capturing the local temporal features of the waveform. Each convolutional layer is immediately followed by the GELU activation function and layer normalization, which is computed as shown in (10):

$$LayerNorm(x) = \gamma \cdot \frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} + \beta \quad (10)$$

where, μ and σ^2 represent the mean and variance of the input x, respectively, ϵ is a small constant added to ensure numerical stability, and γ and β are learnable scaling and shifting coefficients that allow the model to adapt to the normalized outputs during training.

The output of the final convolutional layer is a sequence of latent vectors $z = (z1, z2, \dots z_T)$, which is passed to the context network— a stack of 24 transformer encoder layers with 16 attention heads and a position-wise FFN with a hidden size of 1024. Positional encodings are first added to the sequence, as in the AST model.

Each transformer block contains two sublayers: a multi-head self-attention mechanism and an FFN. The attention mechanism enables each time step to attend to all others, thereby capturing global contextual dependencies. This self-attention mechanism is mathematically expressed in (11):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (11)$$

The FFN in each transformer layer comprises two linear layers separated by a GELU activation, as described in (12):

$$FFN(x) = GELU(xW_1 + b_1)W_2 + b_2 \quad (12)$$

After all transformer layers are processed, the output is a sequence of contextualized embeddings $c = (c_1, c_2, \dots, c_T)$, where each vector $c_t \in \mathbb{R}^d$ encodes the acoustic information at time step t. Instead of using standard mean pooling, this study implements attention pooling to allow the model to learn which time steps are most informative for classification.

This attention mechanism enables the model to focus on the most relevant portions of the cough signal, thereby improving its ability to differentiate between smokers and non-smokers. The attention-weighted vector c_{attn} is passed through a linear classification layer followed by a softmax activation to produce the final prediction, as defined in (13):

$$\sigma(z)_i = \frac{e^{z_i}}{e^{z_0}+e^{z_1}}, i \in \{0,1\} \quad (13)$$

During training, the cross-entropy loss is used to measure the discrepancy between the predicted probabilities and true labels. The loss function $\mathcal{L}$ for the Wav2Vec2 model is defined in (14):

$$\mathcal{L} = -\sum_{i=0}^{1} y_i \log(\hat{y}_i) \quad (14)$$

D. Training Setup

The AdamW optimizer was used to train the models, incorporating a weight decay of 0.001 and an initial learning rate of 1×10^{-4}. Training was conducted for 20 epochs with a batch size of 16. The objective function was cross-entropy loss. Early stopping was applied based on validation loss to prevent overfitting, with a patience of four epochs. Model performance was comprehensively evaluated using several metrics, including accuracy, precision, recall, F1-score, and AUC, based on a 5-fold cross-validation with 75/25 train and validation/test split [22]. All experiments were conducted on an NVIDIA RTX A4000 GPU with 16 GB VRAM.

IV. Experimental Results

A. Evaluation Parameters

This study employed several evaluation metrics, including the accuracy, F1-score, precision, recall, and AUC. The confusion matrix is a fundamental tool for assessing classification model performance, offering a comprehensive comparison between the model's predictions and the actual labels. The confusion matrix comprises four components: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). Precision quantifies the proportion of correctly predicted positive instances among all instances predicted as positive, thereby providing insight into the model's susceptibility to false positive errors. Recall, on the other hand, evaluates the proportion of actual positive instances that the model correctly identified, thereby reflecting its effectiveness in minimizing false negative errors. The F1-score, computed as the harmonic mean of precision and recall, provides a balanced assessment of the model's performance by simultaneously accounting for both FP and FN. Additionally, the AUC, derived from the Receiver Operating Characteristic (ROC) curve, and measures the model's ability to discriminate between classes across various threshold settings. A higher AUC value indicates better overall classification performance, especially in imbalanced dataset scenarios. These evaluation metrics are computed using the formulas presented in (15) through (21).

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (15)$$

$$Recall = \frac{TP}{TP+FN} \quad (16)$$

$$Precision = \frac{TP}{TP+FP} \quad (17)$$

$$F1-Score = \frac{2\times(precision\times recall)}{precision+recall} \quad (18)$$

$$TPR = \frac{TP}{TP+FN} \quad (19)$$

$$FPR = \frac{FP}{FP+TN} \quad (20)$$

$$AUC = \int_0^1 TPR(FPR)dFPR \quad (21)$$

B. Experimental Analysis

In this study, simulations and model development for detecting smokers based on cough sounds, as well as the creation of a digital audio-based detection system, were conducted using hardware and software with specific configurations. Table II presents the detailed specifications of the equipment and tools used.

TABLE II. DEVICE SPECIFICATION

Specifications	
GPU	NVIDIA RTX A4000
GPU Memory	16 GB
RAM	16 GB
Disk	1 TB
Programming Language	Python 3.10

The authors configured the training process with 20 epochs, a learning rate of 0.0001, and a batch size of 16, using the AdamW optimizer. The raw waveform audio was resampled to 16 kHz for input into the Wav2Vec2 model and subsequently transformed into mel-spectrograms with 128 frequency bins for use with the AST model.

As shown in Table III, the Wav2Vec2 model without augmentation achieved the highest overall performance, attaining average accuracy, recall, precision, F1-score, and AUC values of 86.48%, 0.866, 0.868, 0.862, and 0.945, respectively. This model also required the shortest training time, completing each fold in an average of 49 minutes. Although the Wav2Vec2 with augmentation showed comparable AUC values (consistently 0.94 across all folds), its overall performance metrics were slightly lower, and its average training time increased to 65 minutes per fold. On the other hand, the AST model, both with and without SpecAugment, demonstrated moderate performance, with average F1-scores around 0.76-0.77 and longer training times of approximately 78 minutes per fold. Notably, the addition of SpecAugment to the AST did not yield significant improvements and, in some folds, even resulted in minor declines in accuracy. Overall, the Wav2Vec2-based approach outperformed the AST-based method in terms of both predictive performance and computational efficiency, indicating its suitability for fast and accurate detection of smokers based on cough audio.

TABLE III. MODEL'S PERFORMANCE

Model	Fold	Evaluation Parameters					Training Time
		Acc	Rec	Pre	F1-Score	AUC	
AST (w/o Augmentation)	1	76.65 %	0.77	0.77	0.77	0.84	78 min
	2	76.62 %	0.76	0.76	0.76	0.84	78 min
	3	76.49 %	0.76	0.77	0.76	0.85	78 min
	4	78.07 %	0.78	0.78	0.78	0.86	78 min
	5	76.33 %	0.76	0.78	0.76	0.85	78 min
AST (w/ Augmentation)	1	75.75 %	0.76	0.76	0.76	0.84	78 min
	2	77.70 %	0.78	0.78	0.78	0.86	77 min
	3	77.31 %	0.77	0.79	0.77	0.86	77 min
	4	77.52 %	0.78	0.78	0.78	0.85	78 min
	5	76.75 %	0.77	0.77	0.77	0.84	78 min
Wav2Vec2 (w/o Augmentation)	1	87.46 %	0.87	0.87	0.87	0.94	49 min
	2	85.69 %	0.86	0.86	0.86	0.93	49 min
	3	86.54 %	0.87	0.87	0.86	0.94	49 min
	4	86.51 %	0.87	0.87	0.86	0.94	49 min
	5	86.19 %	0.86	0.87	0.86	0.94	49 min
Wav2Vec2 (w/ Augmentation)	1	87.17 %	0.87	0.87	0.87	0.94	65 min
	2	84.53 %	0.85	0.85	0.84	0.94	65 min
	3	86.49 %	0.86	0.87	0.86	0.94	66 min
	4	87.23 %	0.87	0.87	0.87	0.94	65 min
	5	85.35 %	0.85	0.86	0.85	0.94	65 min

V. CONCLUSION AND FUTURE WORK

This study successfully demonstrated that cough sounds can serve as a noninvasive signal for detecting smoking status using deep learning approaches. Among the models evaluated, the Wav2Vec2 architecture without augmentation yielded the best performance, achieving an average accuracy of 86.48%, F1-score of 0.862, and AUC of 0.945, while also requiring the shortest training time of only 49 minutes per fold. This model significantly outperformed the AST-based models, both with and without SpecAugment, which achieved only around 76%–77% accuracy with a longer training duration of approximately 78 minutes per fold. These findings highlight the effectiveness

of raw waveform-based representations like Wav2Vec2 in capturing audio characteristics related to smoking, surpassing the performance of spectrogram-based methods such as AST. Therefore, the raw waveform-based approach using Wav2Vec2 presents a viable and efficient direction for developing automated, non-invasive health screening tools. In practice, this technology could be integrated into mobile applications for the insurance industry, offering an objective method for underwriting and risk assessment that moves beyond self-reported data. In clinical settings, it could serve as a rapid, preliminary screening tool in primary care. However, significant barriers to real-world adoption must be addressed. These include ensuring model robustness against varying audio quality from different consumer-grade microphones, handling background noise, and validating performance across diverse demographic populations.

Future work should focus on these challenges. First, hyperparameter tuning is necessary to further optimize model performance. Second, model ensembling techniques could enhance predictive stability. Finally, and most critically, evaluating cross-dataset generalization is crucial to ensure the model performs reliably on cough samples from different sources and acoustic environments before it can be considered for deployment in real-world scenarios.

ACKNOWLEDGMENT

This research was funded by the 2025 DIKTI Research Grant, under contract number PKS-582/UN2.RST/HKP05.00/2025. The authors also like to acknowledge the support of the Data Science Center FMIPA UI, BrainAI Lab (Bioinformatics Research, Data Intelligence, and AI Innovation Laboratory), PT Seleris Meditekno Internasional, and PT Global Risk Management.

REFERENCES

[1] World Health Organization, “Tobacco.” Accessed: Nov. 12, 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/tobacco

[2] World Health Organization, “Tobacco.” Accessed: Nov. 12, 2024. [Online]. Available: https://www.who.int/europe/health-topics/tobacco

[3] C. B. Sherman, “Health effects of cigarette smoking,” Clin Chest Med, vol. 12, pp. 643–658, May 1991, [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/1747984/

[4] S. P Saha, D. K Bhalla, T. F Whayne, and C. Gairola, “Cigarette smoke and adverse health effects: An overview of research trends and future needs,” International Journal of Angiology, vol. 16, no. 03, pp. 77–83, Sep. 2007, doi: 10.1055/s-0031-1278254.

[5] N. L. Benowitz, K. E. Schultz, C. A. Haller, A. H. B. Wu, K. M. Dains, and P. Jacob, “Prevalence of Smoking Assessed Biochemically in an Urban Public Hospital: A Rationale for Routine Cotinine Screening,” Am J Epidemiol, vol. 170, no. 7, pp. 885–891, Oct. 2009, doi: 10.1093/aje/kwp215.

[6] Z. Ma, C. Bullen, J. T. W. Chu, R. Wang, Y. Wang, and S. Singh, “Towards the Objective Speech Assessment of Smoking Status based on Voice Features: A Review of the Literature,” Journal of Voice, vol. 37, no. 2, pp. 300.e11-300.e20, Mar. 2023, doi: 10.1016/j.jvoice.2020.12.014.

[7] L. Chai, A. J. Sprecher, Y. Zhang, Y. Liang, H. Chen, and J. J. Jiang, “Perturbation and Nonlinear Dynamic Analysis of Adult Male Smokers,” Journal of Voice, vol. 25, no. 3, pp. 342–347, May 2011, doi: 10.1016/j.jvoice.2010.01.006.

[8] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. The MIT Press, 2016.

[9] A. Bustamam, D. Sarwinda, R. H. Paradisa, A. A. Victor, A. R. Yudantha, and T. Siswantining, “Evaluation of convolutional neural network variants for diagnosis of diabetic retinopathy,” Communications in Mathematical Biology and Neuroscience, 2021, doi: 10.28919/cmbn/5660.

[10] A. Salma, A. Bustamam, A. Yudantha, A. Victor, and W. Mangunwardoyo, “Artificial Intelligence Approach in Multiclass Diabetic Retinopathy Detection Using Convolutional Neural Network and Attention Mechanism,” International Journal of Advances in Soft Computing and its Applications, vol. 13, no. 3, pp. 101–114, Dec. 2021, doi: 10.15849/IJASCA.211128.08.

[11] A. Mehrish, N. Majumder, R. Bharadwaj, R. Mihalcea, and S. Poria, “A review of deep learning techniques for speech processing,” Information Fusion, vol. 99, p. 101869, Nov. 2023, doi: 10.1016/j.inffus.2023.101869.

[12] K. Habashy et al., “Cough Classification Using Audio Spectrogram Transformer,” in 2022 IEEE Sensors Applications Symposium (SAS), IEEE, Aug. 2022, pp. 1–6. doi: 10.1109/SAS54819.2022.9881344.

[13] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations,” 2020. [Online]. Available: https://arxiv.org/abs/2006.11477

[14] W.-J. Chung, M. Kim, and H.-G. Kang, “C2C: Cough to COVID-19 Detection in BHI 2023 Data Challenge,” 2023. [Online]. Available: https://arxiv.org/abs/2311.00364

[15] H. Ayadi, A. Elbéji, V. Despotovic, and G. Fagherazzi, “Digital Vocal Biomarker of Smoking Status Using Ecological Audio Recordings: Results from the Colive Voice Study,” Digit Biomark, vol. 8, no. 1, pp. 159–170, Aug. 2024, doi: 10.1159/000540327.

[16] Z. Ma et al., “Automatic Speech-Based Smoking Status Identification,” Lecture notes in networks and systems, pp. 193–203, Nov. 2022, doi: 10.1007/978-3-031-10467-1_11.

[17] A. H. Poorjam, S. Hesaraki, S. Safavi, H. van Hamme, and M. H. Bahari, “Automatic Smoker Detection from Telephone Speech Signals,” 2017, pp. 200–210. doi: 10.1007/978-3-319-66429-3_19.

[18] S. Huddart et al., “Solicited Cough Sound Analysis for Tuberculosis Triage Testing: The CODA TB DREAM Challenge Dataset,” Mar. 28, 2024. doi: 10.1101/2024.03.27.24304980.

[19] Y. C. Eldar, “Resampling,” Cambridge University Press eBooks, pp. 323–367, Dec. 2014, doi: 10.1017/cbo9780511762321.010.

[20] D. Desblancs, “Self-Supervised Beat Tracking in Musical Signals with Polyphonic Contrastive Learning,” May 2022. doi: 10.48550/arXiv.2201.01771.

[21] O. O. Abayomi-Alli, R. Damaševičius, A. Qazi, M. Adedoyin-Olowe, and S. Misra, “Data Augmentation and Deep Learning Methods in Sound Classification: A Systematic Review,” Electronics (Basel), vol. 11, no. 22, p. 3795, Nov. 2022, doi: 10.3390/electronics11223795.

[22] M. Grandini, E. Bagli, and G. Visani, “Metrics for Multi-Class Classification: an Overview,” ArXiv, vol. abs/2008.05756, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:221112671

ProjectNavigator: A Software Project Management Approach Selection Assistant

Lin Dongzhi, Salfarina Abdullah
Department of Software Engineering and Information System-Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Malaysia

***Abstract*—In software projects, project management approaches are crucial. Selecting a suitable management approach based on the specific project characteristics becomes the key to the success of the project. However, software projects are becoming more and more complex, and project managers tend to rely on subjective judgment to select the project management approaches. At present, project managers lack a systematic method or tool that can help them quickly and accurately select the most suitable project management approach to reduce project risks and improve the success rate. The objective of this research is to propose a tool to assist project managers in selecting the most suitable project management approach based on the specific project characteristics. This research will collect and analyze existing project management approaches and their applicable scenarios to extract relevant influencing factors. Then, a recommendation tool is developed to compare and recommend the most suitable project management approach. Finally, the usability and effectiveness of the tool will be validated through expert evaluation and usability testing. Through this tool, project managers can quickly analyze and compare the suitability of different management approaches and obtain specific guidance and suggestions, significantly improving the success rate of projects.**

Keywords*—Software projects; project management; recommendation tool; expert evaluation; usability testing

I. Introduction

With the rapid development of the software industry and rising project complexity, software projects are facing increasing challenges. Different project types and environmental characteristics make it particularly critical to select suitable project management approaches [1]. The most widely used project management approaches are waterfall, agile, and hybrid. Each approach has its unique advantages and limitations. The waterfall approach is known for its structured and phased process, which is suitable for projects with clear and stable requirements, but it has significant limitations in the rapidly changing software development environment [2]. In contrast, the agile approach performs well in projects with frequently changing requirements through iterative development and flexibility, but it may also lead to scope creep and insufficient resource management [3]. In recent years, the hybrid approach has gradually emerged. It combines the planning of waterfall and the flexibility of agile, providing potential solutions for complex projects, but also increasing the complexity of project management [4].

The main challenge for project managers is selecting a suitable management approach that matches project characteristics. Currently, they often rely on subjective judgment, which increases the risk of decision-making [5]. Therefore, they need a tool to assist them in selecting a suitable management approach. However, the existing studies mainly focus on the theoretical analysis of the three approaches and their scenarios or generate recommendation results through static tables [6]. These studies usually have limited analytical dimensions and lack user-friendly tool support, which makes it difficult to meet the needs of comprehensive evaluation for project management approaches.

To address the gap, this study aims to develop a decision-making tool combining comprehensive characteristic analysis and graphical results display. The tool provides intuitive, efficient, and practical support for project managers. It can capture the project characteristics provided by users and combine them with the weight calculation model to generate the recommendation result through a user-friendly interface.

This study focuses on the selection of management approaches for software development projects, specifically waterfall, agile, and hybrid. The target users are project managers and team leaders. To ensure the effectiveness and usability of the tool, this study will validate the tool through expert evaluation and usability testing.

The organization of the paper is as follows. A review of the literature on project management approaches and the key factors influencing approach selection is presented in Section II. Building on the review, Section III introduces the methodology. The results of the evaluation and discussion are given in Sections IV and V, respectively. Finally, Section VI concludes the study and outlines future work.

II. Literature Review

A. Approach and Methodology

In project management, approach and methodology there are two different concepts. The approach is a high-level guidance framework that defines the basic principles and directions of project management. In contrast, the methodology provides specific operational guidance, such as scrum and Kanban. According to Gemino et al. [7], the approach provides strategic guidance on “what to do”, while methodology focuses on the specific implementation of “how to do”. This distinction lays a theoretical foundation for the development of the project management tool in this research. The approach is the primary focus of this research, leaving the specific methodology outside its scope.

B. Waterfall, Agile, and Hybrid Approaches

The waterfall approach is linear and sequential, including five stages: requirements, design, implementation, testing, and maintenance. This approach is particularly suitable for projects with clear requirements and few changes during the project, as it emphasizes detailed planning and thorough documentation. By defining clear stages and verification processes, this approach reduces the project uncertainty. However, it lacks the flexibility to handle changing requirements effectively [8].

The agile approach emerged in the mid-1990s, aiming to solve flexibility and adaptability problems. This approach adopts the iterative development mode, delivering product increments in each iteration of the development process. This ensures timely feedback and rapid adaptation to changing requirements [9]. However, its low reliance on documentation increases the need for effective communication, while its high demand for team autonomy can lead to coordination challenges and scope creep [2].

The hybrid approach combines the advantages of waterfall and agile approaches, avoiding their limitations. The hybrid approach aims to balance flexibility and structure. It ensures stability in project planning while adapting to changes [10]. However, implementing a hybrid approach requires that the team has a comprehensive knowledge of multiple approaches. This understanding is essential to effectively integrate their strengths to achieve project objectives [11].

C. Factors Influencing Approach Selection

The selection of a suitable project management approach depends on multiple contextual factors. To understand which factors are most influential, this study conducted a literature review of 20 recent studies that analyzed or discussed the selection of agile, waterfall, or hybrid approaches. While all 20 studies were reviewed to extract key influencing factors, several representative works are highlighted here to illustrate how these factors affect approach selection.

Thesing et al. [6] proposed a decision model for project management methodology selection, categorizing and summarizing factors around project constraints, time and budget, organizational culture, and team characteristics. In terms of project constraints, projects with high requirement stability and high complexity are suitable for the waterfall approach. The project can be controlled through detailed document management and phasing. In time-sensitive projects, the agile approach performs well with rapid delivery of minimum viable products and a high frequency of customer feedback. Regarding budgeting, the waterfall approach emphasizes detailed cost estimation for projects with fixed costs. Regarding organizational culture, organizations with centralized decision-making and hierarchical management tend to adopt the waterfall approach. In addition, for team characteristics, small teams are easier to implement the agile approach, while large or distributed teams usually rely on the structured management of the waterfall approach.

Ly et al. [12] conducted a detailed analysis of the communication styles in the agile and waterfall approaches, identifying the differences in communication tools and communication channels. The waterfall approach relies on formal communication channels to complete task allocation and progress tracking through structured and standardized tools. For example, the team uses InSite system for task management and issue recording to ensure the traceability of project process and the controllability of plan. In contrast, the agile approach emphasizes more informal communication channels during execution-monitoring-control phase, where 90% of the communication is done through Microsoft Teams and only 10% relies on oral communication. The team can share dynamic information through real-time communication tools and promote task adjustment and feedback cycle. This effectively supports the applicability of the agile approach in the environment with frequently changing requirements and high levels of uncertainty.

The selection of a project management approach is influenced by multi-dimensional factors, including key business drivers, time and resource constraints, stakeholder needs and participation, project risk management, project complexity, and project size and cost [13]. These factors provide a decision-making framework for the evaluation and selection of project approaches. In addition, the hybrid approach shows high adaptability in projects with stable and dynamic requirements, especially for projects with high complexity. The hybrid approach can achieve a balance between structural management and dynamic adjustment, optimize resource allocation and risk management, and is particularly suitable for project environments that cannot be handled by a single approach. In practice, the successful application of the hybrid approach is directly related to the experience of project managers and requires that they dynamically customize and apply specific approaches and tools for each project [14].

Cruz et al. [15] conducted a systematic review of 80 studies comparing waterfall, agile, and lean project management approaches. The study highlighted that the waterfall approach is more suitable for well-defined projects with stable requirements, while agile approach is more adaptable to complexity, technological uncertainty, and frequent delivery. In addition, the study identified that factors such as stakeholder participation, team configuration, and organizational culture play important roles in determining approach suitability.

To provide an overview, Table I summarizes the key influencing factors and their frequency across the 20 studies.

TABLE I FACTORS INFLUENCING PROJECT MANAGEMENT APPROACH

Factors	References																				Frequency
	[7]	[8]	[9]	[2]	[11]	[6]	[12]	[17]	[13]	[14]	[18]	[19]	[20]	[16]	[21]	[15]	[22]	[23]	[24]	[25]	
Requirement Stability	√	√	√	√	√	√	√	√	√	√	√		√	√	√	√	√	√	√	√	19
Complexity	√		√	√	√	√	√	√	√	√	√	√		√	√	√		√	√		17
Time Constraints	√	√	√	√	√	√		√	√		√		√		√						11
Budget Constraints	√	√	√	√	√	√			√		√				√						9
Organizational Culture			√		√	√		√	√	√	√						√	√	√	√	11
Team Characteristics	√	√	√		√	√	√	√	√		√		√		√	√	√	√	√	√	16
Communication Tools							√														1
Stakeholder Participation	√		√	√		√		√	√		√			√	√	√	√	√			12
Business Drivers			√	√		√			√												4
Manager Experience										√	√										2
Contract Type		√									√										2
Innovation Level		√						√	√	√	√	√									6
Technological Uncertainty			√					√		√		√		√	√				√	√	8
Delivery Frequency		√		√	√	√		√		√	√		√		√	√	√	√		√	13
Regulatory Constraints			√	√					√	√	√									√	6

III. METHODOLOGY

A. Identification of Key Factors

To identify the key factors influencing project management approach selection, this study reviewed 20 recent studies. Based on their frequency in Table I, the top 12 most frequently mentioned factors are selected for further analysis and modeling.

B. Construction of the Analytic Hierarchy Process (AHP) Model

AHP is a method for making decisions with multiple criteria. It uses pairwise comparisons to assign weights to each criterion and builds a hierarchy to support decision-making. This makes it useful for selection in complex situations (Amponsah & Amponsah, 2020). In this study, a four-level AHP-based model is developed. It is based on multiple influencing factors and is structured into four levels, as illustrated in Fig. 1:

1) Goal level: Defines the decision objective, which is to select the most suitable project management approach.

2) Category level: Groups the influencing factors into three categories: project-related, organizational, and team-related.

3) Criteria level: Includes the top 12 key influencing factors identified from Table I.

4) Alternative level: Represents the three project management options: waterfall, agile, and hybrid approaches.

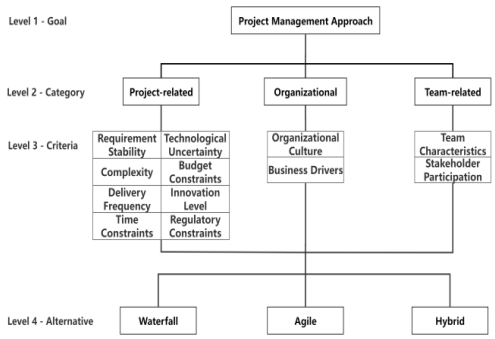


Fig. 1. AHP-based hierarchical model.

C. Pairwise Comparison and Weight Assignment

To quantify the relative importance of each criterion, pairwise comparisons are conducted using the Saaty scale from 1 to 9 [26], as shown in Table II.

TABLE II SAATY'S FUNDAMENTAL SCALE

Intensity of Importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective
3	Moderate importance	Experience and judgment slightly favor one activity over another
5	Strong importance	Experience and judgment strongly favor one activity over another
7	Very strong importance	One activity is strongly favored over another
9	Extreme importance	The evidence favoring one activity over another is of the highest order
2, 4, 6, 8	Intermediate values	Used for a compromise between two adjacent judgments

Within each group of related factors, every item is compared to the others to determine its priority. This process helps assign weights in a clear and structured manner. The comparison matrix for category level is shown in Table III.

TABLE III PAIRWISE COMPARISON MATRIX FOR CATEGORY LEVEL

Category	Project-related	Team-related	Organizational
Project-related	1	4	5
Team-related	1/4	1	2
Organizational	1/5	1/2	1

To ensure the consistency of the pairwise comparison matrix, the Consistency Index (CI) and Consistency Ratio (CR) are calculated. If the CR≤0.1, the matrix is considered consistent. The following formulas are used in the calculation:

$$\lambda_{max} = \sum (A\omega)_i / \omega_i \quad (1)$$

$$CI = (\lambda_{max} - n)/(n-1) \quad (2)$$

$$CR = CI / RI \quad (3)$$

The value of Random Index (RI) [26] corresponding to the matrix size of n=3 can be obtained from Table IV.

TABLE IV RANDOM INDEX VALUE

N	1	2	3	4	5	6	7	8	9
RI	0	0	0.52	0.89	1.11	1.25	1.35	1.40	1.45

For the comparison matrix in Table III, the value of CR is 0.0078. It means that the matrix satisfies the consistency requirement. The weights were calculated using the Principal Eigenvector Method, and the normalized weights are presented in Table V.

TABLE V NORMALIZED WEIGHTS FOR CATEGORY LEVEL

Category	Weight
Project-related	68.3%
Team-related	20.0%
Organizational	11.7%

After determining the category-level weights, pairwise comparisons are conducted within each category to derive local weights for each factor. The global weights are then calculated by combining all the local weights with their corresponding category weights, as shown in Table VI.

D. Recommendation Logic

This study determines the recommendation result by evaluating the alignment between the project characteristics and three project management approaches, including agile, waterfall, and hybrid approaches.

The recommendation tool developed in this study collects project-specific information through a set of guiding questions. Each key factor is associated with 1 to 3 questions. Users are required to rate their project based on these questions using a scale from 1 to 5. The average of all scores under that factor is taken as the final score of this factor.

TABLE VI GLOBAL WEIGHTS OF ALL FACTORS

Category	Category Weigh	Factor	Local Weight	Global Weight
Project-related	68.3%	Requirement Stability	29.9%	20.4%
		Complexity	18.9%	12.9%
		Delivery Frequency	14.2%	9.7%
		Time Constraints	10.7%	7.3%
		Budget Constraints	10.7%	7.3%
		Technological Uncertainty	6.8%	4.6%
		Innovation Level	4.9%	3.3%
		Regulatory Constraints	3.9%	2.7%
Organizational	11.7%	Organizational Culture	75%	8.8%
		Business Drivers	25%	2.9%
Team-related	20.0%	Team Characteristics	66.7%	13.3%
		Stakeholder Participation	33.3%	6.7%

To ensure consistency in calculation, 1–5 scales are normalized to values between 0 and 1. The normalized score for each rating is calculated using the formula:

$$r_i = (s_i - 1)/4 \quad (4)$$

where, s_i is the user's original rating and r_i is the normalized value between 0 and 1.

Next, each normalized score is compared with the ideal value for each project management approach. This study defines the ideal values based on the insight from the literature review. These values reflect the optimal project characteristics. The higher the value, the higher the alignment between this characteristic and the corresponding approach. Table VII presents the ideal values assigned to each factor for the three approaches.

TABLE VII IDEAL VALUES

Factor	Agile	Waterfall	Hybrid
Requirement Stability	0.1	1.0	0.6
Complexity	1.0	0.1	0.8
Delivery Frequency	0.9	0.1	0.6
Time Constraints	1.0	0.3	0.7
Technological Uncertainty	0.9	0.1	0.6
Budget Constraints	1.0	0.3	0.7
Innovation Level	0.9	0.1	0.6
Regulatory Constraints	0.1	1.0	0.8
Organizational Culture	0.9	0.1	0.7
Business Drivers	0.9	0.1	0.7
Team Characteristics	1.0	0.3	0.7
Stakeholder Participation	1.0	0.1	0.7

Following the normalization process, each normalized score is compared with the ideal value of each project management approach. The consistency between the project and a specific approach is quantified by the absolute difference between the user's normalized score and the ideal value. The following formula is used in the calculation:

$$Match_j = 1 - |r_j - p_j| \quad (5)$$

where, r_j is the normalized user rating for the factor, p_j is the ideal value of the corresponding approach and $Match_j$ is the matching score. Each matching score is then multiplied by the global weight of the corresponding factor to reflect its relative importance:

$$WeightedMatch_j = Match_j \times GobalWeight_j \quad (6)$$

The final score of each approach is the sum of the weighted matching values of all factors. For better understanding, the result is multiplied by 100 and presented on a scale from 0 to 100.

$$FinalScore = \sum WeightedMatch_j * 100 \quad (7)$$

The final recommendation result is determined by comparing the final scores across three approaches. The approach with the highest score is recommended as the most suitable approach. Fig. 2 presents a flowchart that summarizes the recommendation process.

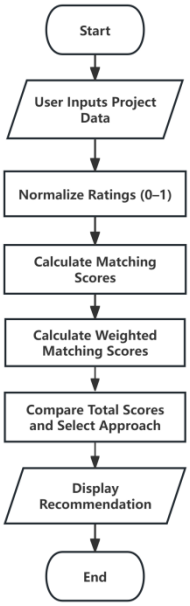


Fig. 2. Flowchart of the recommendation tool.

E. System Implementation

To implement the project management approach recommendation tool, this study develops a lightweight and user-friendly web-based system. The system is built using Spring Boot for backend logic, Thyme leaf for the interface layer, and Chart.js for visualization. It is deployed on the Render platform. This enables real-time user input processing and result display. The system architecture is illustrated in Fig. 3.

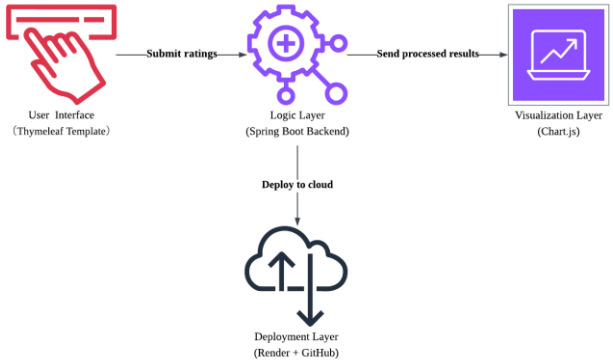


Fig. 3. System architecture of the recommendation tool.

The system provides a bilingual interface and an intuitive layout for users to input project characteristics and receive recommendations. Some snapshots of the interface are shown in Fig. 4 and 5.

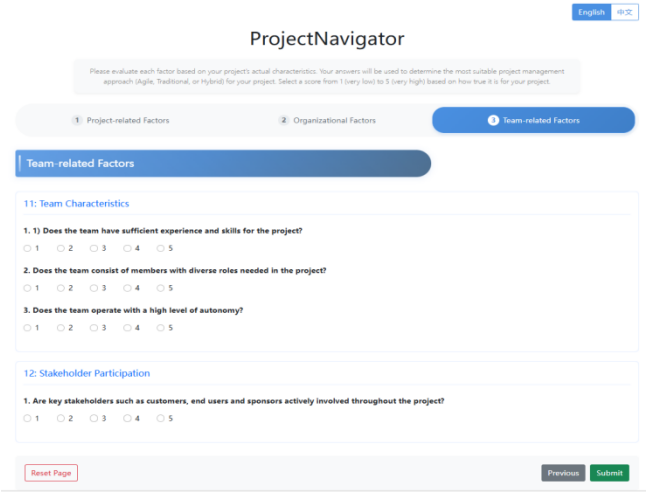


Fig. 4. Team-related questionnaire section.



Fig. 5. Result display area.

F. System Validation

For effectiveness, the factor weights and the recommendation results are examined through expert evaluation. Furthermore, usability testing is carried out using the System Usability Scale (SUS). The results of both validations offer valuable insights for further optimization of the tool.

1) Expert evaluation: To validate the effectiveness of the tool, five experts are invited to complete evaluation questionnaires. These experts have over five years of experience in project management and are familiar with waterfall, agile, and hybrid approaches. Experts need to review the global weights of the 12 factors and evaluate whether each weight is reasonable through a questionnaire. The questionnaire presents the names, global weights, and brief definitions of each factor in a table, ensuring a consistent understanding among experts. For each factor, experts need to evaluate whether the weight is too high, reasonable, or too low based on their professional judgment. If a factor is too high or too low, the system will provide an input box for experts to enter a suggested weight. They were also invited to test the tool and evaluate the accuracy of its recommendation results. Their evaluation focused on whether the recommended approach was accurate and whether they would adopt it in real projects.

2) System usability evaluation: To evaluate the usability of the recommendation tool, five users with different project management experience ranging from one to seven years are invited to complete the SUS questionnaire. SUS is a standardized questionnaire tool consisting of 10 questions. In addition, each question has a Likert scale from strongly disagree to strongly agree and is calculated using standard methods [27]. For usability testing, five users can identify most usability issues [28].

IV. Results

A. Expert Evaluation Results on Factor Weights

Based on the expert evaluation, the global weights of the 12 factors were generally confirmed as reasonable. Only the requirement stability and business drivers were suggested for adjustment by more than three experts. The adjustment details are summarized in Table VIII.

TABLE VIII Adjustments to Factor Weights

Factor	Original Weight	Suggested Value	Average Value	Final Adjusted Weight
Requirement Stability	20.4%	10%,15%,18%,18%	15.25%	17%
Business Drivers	2.9%	5%,5%,7%,10%	6.75%	6%

After adjusting the weights of requirement stability and business drivers, the total weight of all factors is no longer 100%. To maintain consistency, the remaining 10 factors are scaled based on their original weights.

B. Expert Evaluation Results on Recommendation Accuracy

To evaluate the effectiveness of the recommendation results, experts were invited to use the system and provide feedback based on their professional judgment. After using the system, they were asked to evaluate whether the recommended approach was accurate and whether they would adopt it in actual projects. A 5-point Likert scale was used, ranging from strongly disagree to strongly agree. Four experts agreed that the recommended approach is accurate and could be adopted in actual projects. None of the experts chose disagree or strongly disagree.

C. System Usability Evaluation

The final SUS score was calculated as 84.5. According to Hyzy et al. (2022), this score exceeds the benchmark of 68, indicating an acceptable level of usability.

V. Discussion

A. Adjustment of Factor Weights

Based on the expert evaluation, two factors required adjustment: requirement stability and business drivers.

1) Requirement stability: The original weight of requirement stability is 20.4%. According to expert feedback, four experts considered that the value was too high and suggested a range of 10% to 18%. Although two experts suggested 18%, the lowest value of 10% significantly reduced the average to 15.25%. Considering the consensus among experts that this factor is overestimated, the final weight was adjusted to 17%. This adjustment reduces the weight while still maintaining its importance.

2) Business drivers: The original weight of business drivers is 2.9%. According to expert feedback, four experts considered that the value was too low and suggested a range of 5% to 10%. However, because the majority of expert suggestions were below 7%, the final weight was set at 6%. This value is closer to expert suggestions and avoids overadjustment.

B. Recommendation Accuracy

Expert evaluation results on recommendation accuracy indicate a high level of confidence in the effectiveness of the recommended results. This positive feedback validates that this tool can provide reliable and practical recommendation results based on project characteristics.

C. Usability of the System

The SUS score of 84.5 reflects a high level of usability, clearly above the benchmark of 68. This result shows that the tool is easy to learn and use for project managers. High usability also increases the likelihood of adoption in real projects.

D. Validation Significance and Related Work Comparison

These validation measures are essential to ensure that the tool is not only theoretically sound but also practically usable. Unlike many existing studies that only present theoretical models without a practical tool or systematic validation, this study develops the recommendation tool and validates it

through expert evaluation and SUS testing. The results confirm its reliability, recommendation accuracy, and usability, indicating strong potential for adoption in real projects.

VI. Conclusion and Future Work

A. Conclusion

This study developed a web-based recommendation tool to assist project managers in selecting appropriate project management approaches. The tool enables users to input project characteristics through structured questions. Then the system recommends the most suitable project management approach from agile, waterfall, or hybrid approaches.

To ensure the effectiveness and usability of the tool, validation was conducted through expert evaluation and usability testing. Experts evaluated the global factor weights and recommendation results and provided adjustment suggestions for certain weights. Then, the model was optimized based on expert feedback to enhance its consistency with professional judgment. In addition, the system usability was evaluated through the SUS questionnaire, and the final score was 84.5, exceeding the acceptable threshold. The results indicate that the tool has good effectiveness and user experience in assisting the selection of a project management approach.

Overall, the recommendation tool developed in this study offers a clear and reliable way to help project managers select the most suitable project management approach and has strong potential for practical application.

B. Future Work

Although this study has successfully developed and validated the recommendation tool, there are still several areas that can be further improved. To enhance the effectiveness and usability of the tool, the following directions for future improvement are proposed:

- The current system uses the same global weight for all users. In the future, based on providing predefined weight templates, users can adjust these global weights within a limited range according to their own project characteristics. This can enhance the flexibility and personalization of the model while maintaining stability and consistency of recommendation logic.
- In the future, real-world software project data can be collected to further validate the effectiveness of these factors. This would help identify missing factors or weights that may require adjustment.
- A database would be integrated to store user input data and recommendation results. In addition, a feedback module can be added to allow users to evaluate the accuracy of the recommended approach after the project is completed. With sufficient historical data and user feedback, the system can adjust the model, improving the accuracy of the system.

References

[1] V. S. Balasubramaniam, P. R. R. Thumati, P. Kanchi, R. Agarwal, O. Goel, and E. A. Shrivastav, "Evaluating the impact of agile and waterfall methodologies in large scale IT projects," Int. J. Prog. Res. Eng. Manag. Sci., vol. 3, no. 12, pp. 397–412, 2023, doi: https://doi.org/10.58257/IJPREMS32363.

[2] V. Sirisha, V. E. P. J. Sarada, Dr. B. Haralayya, B. V. RamaKrishna, and D. Bisen, "Project Management Methodologies: A Comparative Analysis of Agile and Waterfall Approaches," SSRN Electronic Journal, 2024, doi: https://doi.org/10.2139/ssrn.5000321.

[3] R. Mokhtar and M. Khayyat, "A Comparative Case Study of Waterfall and Agile Management," SAR Journal - Science and Research, pp. 52–62, 2022, doi: https://doi.org/10.18421/SAR51-07.

[4] N. Koceska and S. Koceski, "Hybrid project management as a new form of project management," Journal of Applied Economics and Business, vol. 10, no. 4, pp. 16–23, 2022.

[5] M. J. Bianchi, E. C. Conforto, E. Rebentisch, D. C. Amaral, S. O. Rezende, and R. De Pádua, "Recommendation of project management practices: A contribution to hybrid models," IEEE Trans. Eng. Manag., vol. 69, no. 6, pp. 3558–3571, 2022, doi: https://doi.org/10.1109/TEM.2021.3101179.

[6] T. Thesing, C. Feldmann, and M. Burchardt, "Agile versus Waterfall Project Management: Decision Model for Selecting the Appropriate Approach to a Project," Procedia Computer Science, vol. 181, pp. 746–756, 2021, doi: https://doi.org/10.1016/j.procs.2021.01.227.

[7] A. Gemino, B. Horner Reich, and P. M. Serrador, "Agile, traditional, and hybrid approaches to project success: Is hybrid a poor second choice?," Project Management Journal, vol. 52, no. 2, pp. 161–175, 2021, doi: https://doi.org/10.1177/8756972820973082.

[8] N. Jamous, G. Garttan, D. Staegemann, and M. Volk, "Hybrid project management methods efficiency in IT projects," in AMCIS 2021 Proceedings, [Online]. Available: https://aisel.aisnet.org/amcis2021/it_projmgmt/it_projmgmt/1.

[9] N. Prenner, C. Unger-Windeler, and K. Schneider, "Goals and challenges in hybrid software development approaches," Journal of Software: Evolution and Process, vol. 33, no. 11, p. e2382, 2021, doi: https://doi.org/10.1002/smr.2382.

[10] A. Lalmi, G. Fernandes, and S. B. Souad, "A conceptual hybrid project management model for construction projects," Procedia Computer Science, vol. 181, pp. 921–930, 2021, doi: https://doi.org/10.1016/j.procs.2021.01.248.

[11] J. Reiff and D. Schlegel, "Hybrid project management – a systematic literature review," International Journal of Information Systems and Project Management, vol. 10, no. 2, pp. 45–63, 2022, doi: https://doi.org/10.12821/ijispm100203.

[12] D. Ly, M. Overeem, S. Brinkkemper, and F. Dalpiaz, "The power of words in Agile vs. Waterfall development: Written communication in hybrid software teams," Journal of Systems and Software, vol. 219, p. 112243, 2025, doi: https://doi.org/10.1016/j.jss.2024.112243.

[13] L. H. L. Iwersen, L. Zem, and R. D. A. Penteado Neto, "Choosing the best project management methodology for research and development projects: Agile, Waterfall, or Hybrid?," Revista Foco, vol. 16, no. 11, p. e3336, 2023, doi: https://doi.org/10.54751/revistafoco.v16n11-112.

[14] F. Copola Azenha, D. Aparecida Reis, and A. Leme Fleury, "The role and characteristics of hybrid approaches to project management in the development of technology-based products and services," Project Manag. J., vol. 52, no. 1, pp. 90–110, 2021, doi: https://doi.org/10.1177/8756972820956884.

[15] A. Cruz, A. Tereso, and A. C. Alves, "Traditional, Agile and Lean project management: A systematic literature review," Journal of Modern Project Management, vol. 8, no. 2, pp. 87–95, 2020, doi: https://doi.org/10.19255/JMPM02407.

[16] M. Pace, "A Correlational Study on Project Management Methodology and Project Success," Journal of Engineering, Project, and Production Management, 2019, doi: https://doi.org/10.2478/jeppm-2019-0007.

[17] M. J. Bianchi, E. C. Conforto, and D. C. Amaral, "Beyond the agile methods: A diagnostic tool to support the development of hybrid models," Int. J. Manag. Projects Bus., vol. 14, no. 5, pp. 1219–1244, 2021, doi: https://doi.org/10.1108/IJMPB-04-2020-0119.

[18] B. Székely, S. E. Késmárki-Gally, and Z. Lakner, "Hybrid project management: Scoping review," Project Leadership and Society, vol. 6, p. 100182, 2025, doi: https://doi.org/10.1016/j.plas.2025.100182.

[19] D. Ciric Lalic, B. Lalic, M. Delić, D. Gracanin, and D. Stefanovic, "How project management approach impact project success? From traditional to agile," International Journal of Managing Projects in Business, vol. 15, no. 3, pp. 494–502, 2022, doi: https://doi.org/10.1108/IJMPB-04-2021-0108.

[20] M. Jorgensen, "Relationships Between Project Size, Agile Practices, and Successful Software Development: Results and Analysis," IEEE Software, vol. 36, no. 2, pp. 39–43, 2019, doi: https://doi.org/10.1109/MS.2018.2884863.

[21] University of Houston-Clear Lake, M. Sameen Mirza, and S. Datta, "Strengths and Weakness of Traditional and Agile Processes—A Systematic Review," Journal of Software, vol. 14, no. 5, pp. 209–219, 2019, doi: https://doi.org/10.17706/jsw.14.5.209-219.

[22] S. Mahadik, K. Kodyvaur Krishna Murthy, S. R. Cheruku, A. Jain, and O. Goel, "Agile Product Management in Software Development," International Journal for Research Publication and Seminar, vol. 13, no. 5, pp. 453–467, 2022, doi: https://doi.org/10.36676/jrps.v13.i5.1512.

[23] D. M. Haddab, "Evaluating the Impact of Project Management Approaches on Project Success: A Comprehensive Analysis of Traditional, Agile, and Hybrid Models," Business & IT, vol. XIV, no. 2, pp. 50–60, 2024, doi: https://doi.org/10.14311/bit.2024.02.06.

[24] A. H. Reed, M. G. Angolia, C. Baham and R. Igah, "Usage of Hybrid Project Management Approaches and Influences on Approach Selection," Communications of the Association for Information Systems, vol. 54, no. 1, pp. 402–427, 2024, doi: https://doi.org/10.17705/1CAIS.05415.

[25] B. A. Brahmandam and S. Chandramohan, "An IT project manager's worldwide view on Agile and hybrid methodologies in financial IT," International Journal for Multidisciplinary Research, vol. 6, no. 6, pp. 1–7, 2024, doi: https://doi.org/10.36948/ijfmr.2024.v06i06.43725.

[26] T. L. Saaty and L. G. Vargas, Models, Methods, Concepts & Applications of the Analytic Hierarchy Process, vol. 175. Springer US, 2012, doi: https://doi.org/10.1007/978-1-4614-3597-6.

[27] M. Hyzy, R. Bond, M. Mulvenna, L. Bai, A. Dix, S. Leigh, and S. Hunt, "System Usability Scale Benchmarking for Digital Health Apps: Meta-analysis," JMIR mHealth and uHealth, vol. 10, no. 8, e37290, 2022, doi: https://doi.org/10.2196/37290.

[28] G. Delnevo, V. Ghini, E. Fiumana, and S. Mirri, "A support tool for emergency management in smart campuses: Reference architecture and enhanced web user interfaces," Sensors, vol. 24, no. 18, p. 5887, 2024, doi: https://doi.org/10.3390/s24185887.

A Hybrid RoBERTa-BiGRU-Attention Model for Accurate and Context-Aware Figurative Language Detection

Dr. Sreeja Balakrishnan[1], Rahul Suryodai[2], S. Manochitra[3], Jasgurpreet Singh Chohan[4], Karaka Ramakrishna Reddy[5], A. Smitha Kranthi[6], Dr Ritu Sharma[7]

Assistant Professor (SS), Department of English, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore-641043, Tamil Nadu, India[1]

Senior Data Engineer (Data Governance, Data Analytics: Enterprise Performance Management, AI & ML), USA[2]

Assistant Professor, Department of Computer Science and Engineering, St. Joseph's College of Engineering, Chennai, India[3]

Marwadi University Research Center-Department of Mechanical Engineering-Faculty of Engineering & Technology, Marwadi University, Rajkot, Gujarat, India[4]

Assistant Professor, Department of BS & H, B V Raju Institute of Technology, Narsapur, Medak, Telangana, India[5]

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, AP-522302[6]

Assistant Professor, Department of Mathematics and Humanities, M. M Engineering College, MMDU Mullana, India[7]

***Abstract*—Figurative language, especially sarcasm, poses strong challenges for Natural Language Processing (NLP) models because of its implicit, context-sensitive nature. Both traditional and transformer models tend to find it difficult to identify these subtle forms, particularly when dealing with imbalanced datasets or without mechanisms for targeted interpretability. For overcoming these shortcomings, this study recommends a hybrid deep learning architecture that integrates RoBERTa for high-contextual embedding, Bidirectional Gated Recurrent Units (BiGRU) to capture bidirectional sequential relations, and an attention mechanism allowing the model to focus on the most informative parts of the input text. This integration enhances semantic understanding and classification accuracy compared to current solutions. The model is trained and tested on the benchmark News Headlines Dataset for Sarcasm Detection using binary cross-entropy loss minimized with Adam, along with dropout and learning rate scheduling to avoid overfitting. Experimental results yield better performance, attaining an accuracy of 92.4%, a precision of 91.1%, a recall of 93.2%, and an F1-score of 92.1%. These results outperform baseline techniques such as BiLSTM with attention and fine-tuned BERT variants. Implementation uses PyTorch and Hugging Face Transformers, ensuring reproducibility and extensibility. While effective, the model faces challenges with figurative expressions requiring external world knowledge or cultural context beyond pretrained embeddings. Future work aims to integrate external knowledge graphs and extend the model to multilingual and cross-domain scenarios. This hybrid framework advances figurative language detection, contributing to the broader goal of enhancing AI's nuanced understanding and interpretability of human language.**

***Keywords*—Figurative language detection; sarcasm classification; RoBERTa-BiGRU-Attention model; contextual embeddings; Natural Language Processing**

I. Introduction

Figurative language, including types like sarcasm, metaphor, and irony, is critical in human communication as it communicates beyond the literal meaning [1]. It is particularly dominant in digital text, such as social media, online reviews, and newspaper headlines [2]. These expressive mechanisms are used to add subtlety, signal humor, or highlight perspectives; yet their inherent implicitness and extreme dependency on context make them an overwhelming problem for Natural Language Processing (NLP) algorithms [3]. Traditional NLP models usually find it difficult to correctly infer nonliteral meaning, resulting in misclassification and misinterpretation [4]. This shortcoming has a negative effect on most downstream applications that are dependent on the correct semantic interpretation, including sentiment analysis, opinion mining, emotion detection, and conversational AI. For example, failure in sarcasm detection can invert the targeted sentiment of text, resulting in the wrong analysis outcomes and inferior user experiences.

Emergent developments of models such as BERT and RoBERTa have significantly progressed many NLP tasks through their capacity for creating contextual embeddings capturing syntactic and semantic relations in texts [5]. In spite of these developments, such models still struggle with figurative and subtle language [6]. Such models tend to fail to capture fine semantic signals and long-distance dependencies necessary for distinguishing figurative expressions [7]. Furthermore, issues like severe class imbalance in training sets restrict model generalization [8]. Another major flaw is the lack of comprehensive explainability for transformer-based predictions, which restricts knowledge of why a certain input was labeled as figurative or not, hence influencing trust and usability in actual systems.

Against such challenges, this work suggests a new hybrid framework that is tasked with enhancing the accuracy of figurative language detection through the use of complementary deep learning methods. The model uses RoBERTa to produce high-contextualized word embeddings, a (BiGRU) to capture

bidirectional sequential dependencies in the text, and an attention mechanism to attend to the most contextually informative tokens in an input sequence. By combining these parts, the model hopes to harness the contextual capability of transformers with the added sequential learning and focus prioritization, achieving more accurate and context-aware identification of figurative language. This should yield more accurate outputs that can be used in practical NLP applications that make use of subtle language comprehension.

A. Research Motivation

Figurative language, like irony, metaphor, and sarcasm, is important in the way humans convey meanings that transcend literal understandings. Figurative language in digital communication, particularly on social media and news websites, is prevalent and powerful. Yet, conventional NLP systems often misinterpret figurative language because of its contextual subtlety and implicit semantics. This results in tremendous challenges for applications such as sentiment analysis, emotion detection, and user interaction in AI applications. Despite the progress made by models such as BERT and RoBERTa, precise identification and interpretation of figurative language remains a challenging task. Current models lack class balance, explainability, and proper emphasis on sequential context. Thus, it is an urgent need to create stronger and contextualized approaches that can better represent the richness of figurative language. This work is stimulated by the desire to enhance automatic detection of figurative expressions through using hybrid deep learning models that combine strong contextual embeddings with sequential and attention mechanisms, improving at last performance and explainability of figurative language detection systems.

B. Research Significance

Automatic identification of figurative language is of tremendous importance in the development of NLP applications, such as sentiment analysis, conversational AI, content moderation, and recommendation systems. Incorrect interpretation of figurative language—sarcasm, especially—can cause errors in user intent, resulting in ambiguous sentiment classification or unsuitable responses for AI systems. It is a critical challenge to overcome in improving user experience and decision-making in automated services. Through the creation of a new hybrid architecture consisting of RoBERTa contextual embeddings, Bidirectional Gated Recurrent Units (BiGRU), and an attention mechanism, this work immensely helps bridge the gaps in current methods that fail to capture semantic depth and contextual subtleties. The approach enhances classification accuracy and generalization on benchmark data sets as well as facilitates improved explainability of decisions via attention mechanisms. The adaptability of the work to multilingual and cross-domain settings underscores its wider significance beyond English text processing. This research, therefore, is a significant step towards more human-like and intelligent language comprehension in artificial intelligence systems, with various applications in cross-sections of fields demanding subtle linguistic analysis and interpretation.

C. Key Contribution

- Proposed a novel hybrid deep learning architecture integrating RoBERTa, BiGRU, and attention mechanisms for effective figurative language detection.
- Demonstrated superior performance metrics on benchmark sarcasm detection datasets compared to baseline methods.
- Addressed challenges of context sensitivity and class imbalance to improve model generalization and robustness.
- Enhanced explainability of model predictions through the integration of an attention mechanism highlighting important tokens.
- Provided a framework adaptable to multilingual and cross-domain applications, expanding the utility of figurative language detection in NLP.

The remainder of the section is prepared as follows: Section II is the related study, Section III is the Problem Statement, methodology part is in Section IV, Section V is the result and discussion section, Section VI is the conclusion and future work section.

II. Related Works

Liu et al. [11] explored the issue of nonliteral language and presented Fig-QA, a Winograd-like question-answering dataset that assesses the linguistic models on how well they comprehend the figurative language with contrasting semantics. The task consists of choosing the appropriate meaning between paired figurative expressions whose contextual suggestion varies, and, thus, challenges the profundity of nonliteral understanding of language models. The study compared the functioning of a number of state-of-the-art transformer-based models on the Fig-QA benchmark. Despite these models showing levels of performance far above-average random chance, they were still very far behind the performance of human participants in both zero-shot and few-shot settings. Such a gap highlights the inability of current models to reflect the fine and easily context-specific details of the figurative language. The results indicate that current neural networks, despite their impressive literal comprehension ability, are not very advanced in reasoning, a prerequisite to effective nonliteral comprehension.

Chakrabarty et al [12] designed DREAM-FLUTE, a new framework of figurative language understanding using scene elaboration and entailment inference to increase the accuracy of the interpretation. Metaphors and similes are often used as figurative language, and the sense therefore is not inferred easily when read their surface form. In order to solve this, the authors posed a hypothesis that the proper interpretation will involve building a mental model of the described situation. DREAM-FLUTE builds on this hypothesis by creating a well-organized mental model of both the premise and the hypothesis based on an existing framework known as DREAM (Deep Reading for Elaborative Mental Modeling). Then it produces entailment or contradiction judgments and produces textual explanations. The system was tested as part of the FigLang2022 Shared Task, in which it topped the leader board (tied in first place with 63.3%

Acc@60 accuracy). The authors also showed that by using ensemble methods to improve the performance of a system further was even possible.

Huang et al. [13] reduced the various limitations of existing benchmarks of figurative language. Although the use of figurative language was in the past successfully put within the framework of the Recognizing Textual Entailment (RTE) paradigm, much earlier benchmarking efforts were hampered by spurious correlation and annotation artefacts, and it is hard to assess whether language models are understanding nonliteral meaning effectively. FLUTE bridges this divide by presenting 9,000 pieces of real NLI examples together with human-generated clarifications on the same. The data consists of four categories of figurative language comprising sarcasm, simile, metaphor, and idioms. It was built in the model-in-the-loop fashion, integrating the power of GPT-3 with crowdsourcing and human expert annotation in order to achieve quality and scale. The authors have proven that such a hybrid annotation approach contributes greatly to the reasonable development of complicated linguistic databases. Baseline tests implemented with a fine-tuned model T5 on FLUTE showed that the presence of textual explanations gives an advantage to models, meaning a better understanding of answers and something to be noticed about their possible better performance in figurative situations.

Sharma et al [14] explored transformer-based architectures in identifying figurative presentations compared to literal utterances in huge textual information. With the growing demand for efficient sentiment extraction and text classification with an understanding of language specificities, the authors found it necessary to run a set of transformer models (including BERT and RoBERTA variants), whereas the latter could efficiently perform sentiment extraction and many aspect specifications when it came to nonliteral language. The authors used the opportunity to determine whether the models that were trained on one category of figurative language representation, e.g., metaphors or sarcasm, could generalize well across categories. The study has extensively experimented to indicate that transformer models have achieved significant advances in typical NLP tasks, but are still not very successful with figurative constructs in NLP since they have implicit and context-sensitive definitions. The study also highlighted the urgency with which machines should be made to comprehend the figurative language that is rampant in the human language, and unlike literal language, which is processed.

Kumar et al. [15] approached the problem of interpretation of affective dimensions in the context of conversational dialogue, which are frequently masked by the use of the figurative language of irony, sarcasm, and metaphor. In order to better understand such situations, the authors have suggested a deep neural network model (MOSES) that could be applied to Sarcasm Explanation in Dialogues (SED). The model uses input multimodal sarcastic utterances and outputs natural language explanations to disclose an ironic meaning behind it. This direct encoding of figurative purpose is in turn used to enhance a number of downstream NLP tasks, such as detecting sarcasm, identifying humor and detecting emotion. The tests on different experimental conditions proved that MOSES shows superior results by 2% to the state-of-the-art in the SED in terms of ROUGE, BLEU, and METEOR. Moreover, the inclusion of generated explanations managed to obtain a significant 14% boost in sarcasm detection tasks, and a 2% boost in both humor and emotion recognition tasks.

Vitiugin and Paakki [16] proposed an effective framework regarding the classification of such a complicated task, the Deep Ensemble Soft Classifier (DESC). In view of the fact that the figurative language forms have a complex nature of figurative language use, especially sarcasm, irony, and metaphor, the present study sought to traverse through the inconsistent semantics that are usually intertwined with an expression of a figurative language form. The first contribution of the authors was the presentation of a complete data preprocessing pipeline and its use to standardize and optimize the input representations to deep learning models. They further derived some specialized features like the syntactic structure, mode of expressing emotions, expressive style, and temperamental formulations in the articulation of a figurative meaning, as held by those utilizing social media. Such dense feature representations were next input into the DESC model, an ensemble scheme that mixes different deep learning structures together to boost the accuracy of classification and generalization. Tests have been done using three benchmark sets, with one made particularly in multiple forms of figurative language.

Hülsing and Im Walde [17] provided a unification of a framework that is able to detect multiple figures of speech in more than one language. Aware of the fact that figurative meanings, i.e., metaphors, sarcasm, and idioms, are not solely linguistically diversified, but also culturally conditioned, the study implemented the multilingual multi-figurative language modeling. The authors came up with a sentence-level corpus of three varieties of figurative language and seven languages with the aim of eliminating the deficit of existing monolingual and figure-specific studies. They use template-based prompt learning, which was used to facilitate the model to generalize through figures of speech and languages, with no need to add specialized modules. Such a coherent system reduces the complexity of modeling and allows greater scalability in low-resource or multilingual settings. Experimental studies showed that the proposed technique feasibly outperforms growing transformer-based baselines, which indicated the potential capability of the approach under study in robust cross-language figurative language understanding.

Ocampo and Clarifiño [18] considered the metaphor detection task in figurative language processing to come up with a new approach that uses lexical definitions to boost metaphor detection, hence MIss RoBERTa WiLDe. This is especially true since the inception of the powerful pretrained language models, GPT-3, XLNet, and BERT, which have allowed challenges restricted to small quantities of annotated data, like metaphor recognition, to improve considerably. The authors extend this development by utilizing an open-source lexical database, known as Wiktionary, in to access automatically via dictionary on non-figurative word senses. The definitions that were put forward are contextual anchors to assist the model to differentiate between uses that are literal and uses that are metaphorical by having a contrast between basic use and contextual use. The suggested model combines RoBERTa and a binary token-level classifier to find metaphorical expressions, and it shows a high degree of transferability across datasets.

Experimental comparisons indicate that MIss RoBERTa WiLDe performs end-to-end better or equally well to the state-of-the-art metaphor detection systems, providing a scalable, dataset-independent, and efficient approach.

Lv et al. [19] proposed RB-GAT, a first hybrid text classification model focusing on a well-integrated RoBERTa and BiGRU with Graph Attention Networks (GAT) to solve the weakness of the traditional graph-based neural models in modeling the contextual semantics in a text. Graph Neural Networks (GNNs) have shown promising results on textual classification but tend to fail at capturing text sequential dependencies and contextual information (at the document level). In a bid to overcome this, RB-GAT uses RoBERTa to produce rich but contextualized word and sentence-level embeddings, which are then coupled with a (BiGRU) used to model long-range and bidirectional textual relationships. Such added features are subsequently multi-head GAT, which learns inter-token relations, where the various sections of the document graph are designated by dynamic scores of attention. This kind of arrangement enables the model to be semantically and topologically coherent. The softmax layer is then used in the final classification. Comparing 5 benchmark datasets-Ohsumed, R8, MR, 20ng, and R52, RB-GAT was superior to the baseline model, having achieved an accuracy of 71.48, 98.45 and 80.32 and 90.84, and 95.67, respectively.

Berger et al [20] employed transfer learning in the German setting on a metaphor-recognition task and accordingly used a framework that was cross-lingual, where an English corpus with the sentence-level metaphor is annotated. The English answers were mechanically translated into German, and a sample of 1,000 German sentences was manually parsed in order to become a gold standard of two assessment configurations: sequence labeling composed of tokens and sentence classification. There were two major approaches to transfer learning that the authors tested: 1) transformer-based models, including multilingual BERT, and 2) bilingual word embeddings with an RNN-based classifier. Transformer models performed fairly in zero-shot, as they produced the best performance of 61% F1-score used to perform the classification tasks, whereas the embedding-based approach did not perform any better compared to a random baseline, scoring a mere 36% F1. Once fine-tuned using the annotated German dataset, both methods increased greatly in performance, with a maximum F1-score of 90%, and so cross-lingual metaphor detection can be highly efficient in terms of language-specific tuning. A rich annotated metaphor corpus in the two languages forms a contribution of the study that can be applied to make future studies on mixed language processing of the figurative language more feasible.

Current literature emphasizes the increasing need for advanced models capable of dealing with various kinds of nonliteral semantics in languages and different types of figurative language. Benchmarks such as Fig-QA and FLUTE show that while transformer models are better than chance, their understanding of figurative language is constrained in comparison to human inference. Models such as DREAM-FLUTE and MOSES show that explanation generation and mental modeling together increase interpretability along with performance. Other methods such as DESC and RB-GAT combine contextual embeddings with deep learning models (BiGRU, GAT), enhancing classification accuracy over diverse figurative expressions. Additionally, studies on template-based prompt learning and multilingual embeddings yield high-performance results in cross-lingual metaphor detection, pointing toward a shifting future for strong, scalable processing of figurative language.

III. Problem Statement

Precise identification of figurative language, especially sarcasm, continues to be a tough task in (NLP) because it is implicit, context-dependent, and nuanced [9]. Although models such as BERT and RoBERTa have pushed the frontiers of many NLP tasks further, they tend to struggle with properly capturing subtle semantic signals and long-distance dependencies required for first-order interpretation of nonliteral expressions [10]. Besides, current methods are challenged by class imbalance problems in training data, weak explainability of model choices, and weak infusion of sequential context, all of which lower their overall performance and dependability. This constraint negatively impacts many downstream NLP tasks like sentiment analysis, opinion mining, conversational AI, and emotion detection, wherein the misunderstanding of figurative language can result in inaccurate outcomes and less-than-optimal user experiences. This study resolves the issue of creating a robust, context-sensitive system for figurative language identification that surpasses the limitations of existing approaches. In particular, it introduces a new hybrid model synergizing the benefits of transformer-based contextual embeddings and sequential learning through Bidirectional Gated Recurrent Units (BiGRU) and an attention mechanism. This integration has the goal of enhancing accuracy, interpretability, and generalizability across various kinds of figurative language, eventually enhancing the reliability and usability of figurative language detection in practical NLP applications.

IV. Proposed Hybrid Framework for Figurative Language Detection

A hybrid deep learning architecture is created through the integration of transformer-based contextual embeddings, sequential learning, and attention mechanisms to enhance the identification of figurative language. The architecture incorporates pretrained RoBERTa embeddings, which produce high-quality and context-aware representations of words, and a Bidirectional Gated Recurrent Unit (BiGRU) that learns sequential dependencies in both directions. An attention mechanism is added to emphasize and concentrate on the most informative tokens of the input sequence, enhancing the model's capability to distinguish subtle figurative uses like sarcasm. It is trained and tested on the News Headlines Dataset for Sarcasm Detection, a common benchmark with balanced sarcastic and non-sarcastic samples. Preprocessing of data involves conversion of headlines to lowercase, removal of special characters, and tokenization with RoBERTa's tokenizer. The input is padded or truncated to a constant length to have a consistent input size for batch processing. Training is done with binary cross-entropy as the loss function, optimized using the Adam optimizer. Hyperparameters like learning rate, batch size, dropout rate, and epochs are specifically tuned to minimize overfitting and maximize generalization. The use of a dropout

layer keeps overfitting at bay, while learning rate scheduling stabilizes the convergence process. To holistically quantify model performance, standard classification metrics are provided. This framework of evaluation ensures balanced quantification of the model's capacity to correctly identify figurative language versus literal expressions, ultimately enhancing the robustness and usability of the detection of figurative language in practical applications of NLP. The end-to-end architecture that is proposed is a decent balance between complex semantics understanding as well as sequence-based learning, and would lead to clean and accurate detection of figurative language. Fig. 1 is a pictorial representation of it.

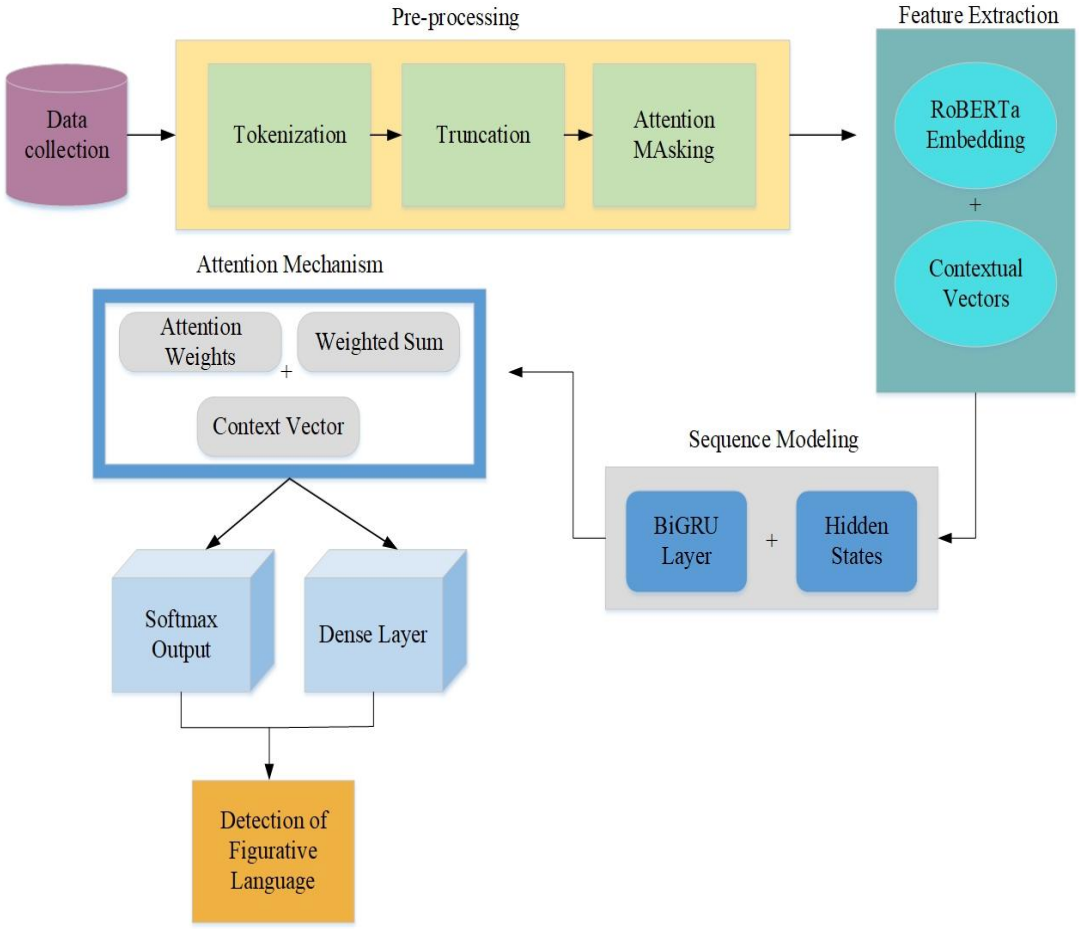


Fig. 1. Proposed RoBERTa-BiGRU-attention architecture

A. Data Collection

The News Headlines Dataset [1] for sarcasm detection, available on Kaggle and developed by Rishabh Misra, is utilized in this research [21]. The dataset comprises approximately 28,000 English-language news headlines, each labeled as either sarcastic (1) or non-sarcastic (0). Each entry contains the headline text, its corresponding sarcasm label, and in some cases, links to the original news articles. This dataset is well-suited for figurative language detection, particularly sarcasm, as it consists of concise texts with sufficient contextual information to train deep learning models effectively. The headlines exhibit clear syntactic structure and provide a clearly labeled, supervised learning environment, enabling rigorous evaluation of the model's ability to comprehend situations and interpret nonliteral language. Table I shows the dataset description.

TABLE I. DATASET DESCRIPTION

Label	Description
0	Non-Figurative Headlines
1	Figurative (Sarcastic) Headlines
Total	50,000

B. Data Pre-Processing

All headlines were in lower case, lacking punctuation, figures, and unnecessary symbols, so as to normalize the input. A fixed-length L (RoBERTa) WordPiece tokenizer was used to tokenize cleaned text and truncate or pad it to the desired length. Subsequent to this, the lean pipeline minimizes vocabulary sparsity, and the input size is maintained effectively to train

[1] https://www.kaggle.com/datasets/rmisra/news-headlines-dataset-for-sarcasm-detection

it. Formally, the cleaned headline x_{clean} is obtained from the original headline x as:

$$x_{clean} = Clean(x) \quad (1)$$

In Eq. (1), x is the original headline whereas x clean is the cleaned version in. A small vocabulary size is achieved through such standardization and makes learning easier and faster. And to each cleaned headline x clean, a Word Piece tokenizer that works with RoBERTa is used to tokenize each. Tokenization transforms a sentence into a sequence of tokens:

$$T = Tokenizer(x_{clean}) = [t1, t2, \ldots, tn] \quad (2)$$

In Eq. (2) $T = \{t1, t2, \ldots, tn\}_0$ refers to the order of subword tokens. The input sequences are of variable length and, as such, are padded or truncated to a set maximum length L, so that the shape of the inputs to a batch processing is the same shown in (3):

$$T' = \frac{pad}{truncate(T,L)} = [t_1, \ldots., t_l] \quad (3)$$

Finally, tokens are converted into embeddings using RoBERTa's pretrained encoder:

$$E = RoBERTaEncoder(T') = [e_1, e_2, .., e_l] \quad (4)$$

In Eq. (4) $E \in \mathbb{R}^{L \times d}$ is the word embeddings of each token, and d is the embedding dimension. This pre-processing chain is designed to yield a clean, tokenized, semantically enriched text to achieve good training and classification.

C. Model Architecture: RoBERTa-BiGRU-Framework

The suggested method for figurative language detection uses a hybrid model with deep learning algorithms, combined with contextual embeddings and sequence learning. Following preprocessing, every headline is tokenized and fed into a pre-trained RoBERTa model, which produces contextual embeddings that have the ability to sense minute semantics and relationships between tokens and thus can effectively identify figurative language patterns in the text.

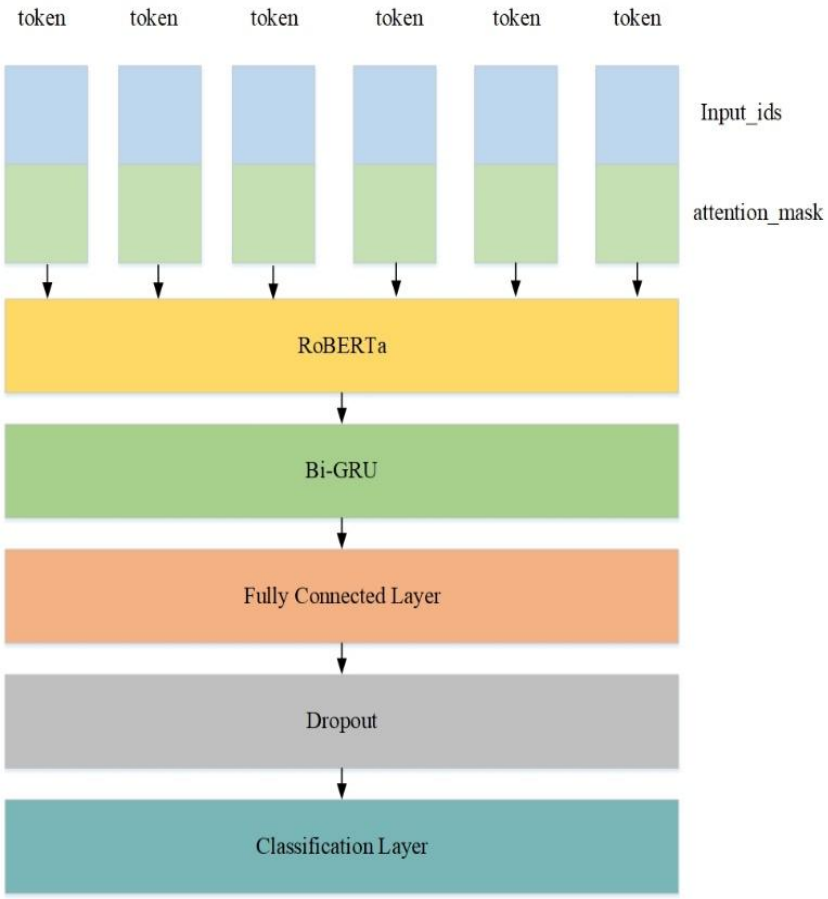


Fig. 2. Block diagram of the hybrid model pipeline.

Fig. 2 represents the proposed hybrid model intended to detect the figurative language in textual data based on the option of RoBERTa and BiGRU. The model takes as an input sentence which is translated into tokens and some other input_ids and attention_mask. The token representations will be input into the RoBERTa module, a pretrained transformer language model, and produce rich contextual representations of tokens capturing semantic and syntactic aspects. RoBERTa output embeddings are passed as input to a Bidirectional Gated Recurrent Unit (BiGRU) network that performs the further processing of the sequence in both forward and backward directions. This dual-processing capacity enables the model to keep the contextual relationships of previous and subsequent tokens and improve expression of linguistic structure and meaning. After output of

the BiGRU, the Fully Connected Layer condenses the sequence-level features into a categorical condensed feature more appropriate to give to the classifier. To avoid overfitting, the model feature contains Dropout Layer that eliminates a random selection of connections in the training process. The output of that processed information is fed to the Classification Layer, which uses the Softmax function to make the prediction in regards to what type of label is in the data figurative (1) or non-figurative (0). This architecture is highly integrated in terms of both contextual, sequential and classification in the recognition of the figurative languages.

Officially, the embed output may be well formulated as: $E \in \mathbb{R}^{L\times d}$, where L is the length of the sequence and d is the embedding dimension. The latter embeddings are taken as input into a (BiGRU) to learn forward and backward dependencies throughout the sequence, which conveys more of those contextual cues. The hidden state over time t is calculated as:

$$h_t = BiGRU(E_t), t = 1,2, \dots, L \quad (5)$$

In Eq. (5), to enhance interpretability and focus on key parts of the sentence, an attention mechanism is applied to the BiGRU outputs. This mechanism assigns higher weights to more informative tokens. The attention weight αt is calculated as:

$$\alpha_t = \frac{\exp(u_t)}{\sum_{j=1}^{L}\exp(u_t)} \quad (6)$$

In Eq. (6), $e_t = tanh(W_h h_t + b$ The final attended representation is a weighted sum of hidden states:

$$r = \sum_{t=1}^{L} \propto_t h_t \quad (7)$$

In Eq. (7), this vector is passed through a fully connected classification layer with a Softmax function to output the probability of the sentence being figurative (e.g., sarcastic):

$$y = Softmax(W_c r + b_c) \quad (8)$$

In Eq. (8), the model is trained using binary cross-entropy loss and optimized with the Adam optimizer. Dropout and learning rate scheduling are employed to prevent overfitting and ensure generalization.

D. Training Parameters and Hyperparameters

The trained model uses the binary cross-entropy loss, optimized by the Adam optimizer to better reduce classification mistakes. Major hyperparameters were finely tuned to achieve the best results on the task of detecting sarcasm. The pretrained RoBERTa-base model is used as the embedding backbone, with a maximum sequence length of 128 tokens to strike the right balance between computation and context attainment. A learning rate of 2e-5 was chosen following initial experiments, yielding stable convergence during training. The batch size is fixed at 32 to ensure full utilization of the GPU without memory overflow. To avoid overfitting, a dropout rate of 0.3 is utilized with random disabling of network connections during training. The model is trained for 10 epochs, and learning rate scheduling is used to dynamically adjust the learning rate and enhance training stability. All selected hyperparameters and configurations are listed in Table II. These parameters serve as the basis for strong and accurate figurative language classification with good generalizability to various inputs.

TABLE II. TRAINING PARAMETERS AND HYPERPARAMETERS

Parameter	Value
Pretrained Model	RoBERTa-base
Sequence Length	128 tokens
Optimizer	Adam
Learning Rate	2e-5
Dropout Rate	0.3
Batch Size	32
Epochs	10
Loss Function	Binary Cross-Entropy

Table II outlines the key training configurations and hyperparameters used for building the proposed model. These settings were tuned to optimize performance on the sarcasm detection task.

Algorithm 1: Figurative Language Detection Using RoBERTa-BiGRU-Attention

```
Input: Headline Text
Output: Label (0 = Non-Figurative, 1 = Figurative)
Begin
  If Headline Text is empty, then
    Return "Invalid input."
  Else
    Clean Text ← Preprocess (Headline Text)
    If Clean Text contains special characters, then
      Remove special characters
    End If
    Tokens ← Tokenize (Clean Text)
    Embeddings ← RoBERTa(Tokens)
    If Embeddings is not None, then
      BiGRU Output ← BiGRU(Embeddings)
    Else
      Return "Embedding failure"
    End If
    Attention Weights ← Compute Attention (BiGRU Output)
    Context Vector ← WeightedSum (BiGRU Output, Attention Weights)
    Prediction ← Softmax (Dense (Context Vector))
    If Prediction ⩾ 0.5 then
      Return 1 Figurative
    Else
      Return 0 Non-Figurative
    End If
  End If
End
```

Algorithm 1 proposes a step-by-step procedure for identifying the presence of figurative language based on a hybrid deep learning model composed of RoBERTa, BiGRU, and an Attention Mechanism. This process starts with an input text that may be taken as a headline. In the case where the input is empty, the algorithm directly responds as invalid. Other than this, other preprocessing is done, such as cleaning up, lowercasing, and taking away any special characters to standardize the form of input. Then the cleaned text is tokenized into the subword units that could be addressed by the RoBERTa model. RoBERTa

model produces contextual representations, which retain the linguistic sensible in the individual words. The embeddings are fed to a Bidirectional Gated Recurrent Unit (BiGRU), which learns forward and backwards dependencies in the text, which is why this is perfect at processing text that is difficult to process, such as the sentence structure of figurative language. When valid embeddings are produced, an attention mechanism is used to highlight the most informative sections of the BiGRU output. A dense layer and a Softmax classifier are then used to consume this weighted combination of the outputs, and this is known as the context vector. The model classifies the input as either non-figurative (0) or as figurative (1) based on the resulting probability score. This architecture is step-by-step in the way that it guarantees robust context-sensitive figurative language detection.

E. Evaluation Metrics

The performance of the suggested model is measured using a variety of typical classification metrics. Precision indicates how accurately the model can detect instances of figurative language out of all instances it detects as figurative, which is a measure of its reliability in detection. Accuracy is the overall precision of the predictions made by the model on the entire dataset, which reflects how well the model differentiates between figurative and non-figurative language at a general level. Recall gauges the effectiveness of the model in identifying all real instances of figurative language in the text, emphasizing its completeness. The F1-score amalgamates precision and recall into one measurement, reconciling the trade-offs between false positives and false negatives, and offering an overall measure of model performance. Because figurative language is so subtle and context-specific, these measurements as a whole provide an in-depth measurement of the model's capability to identify subtle expressions accurately. Evaluation is done solely on a test set that is independent of the training data to guarantee unbiased and sound performance measurement.

Fig. 3 shows the end-to-end procedure of the figurative language detection with the help of a hybrid RoBERTa-BiGRU-Attention model. It begins by accepting a headline text entry. Next, an initial validation will be applied that will clean the input, whether it is empty or not. In case of an empty input, the system would send a response message of invalid input and end. Other than that, the text is preprocessed by converting to lowercase and excluding unwanted special characters. The scrubbed text is tokenized and fed to a pretrained RoBERTa, completing a rich contextual embedding. In case the RoBERTa does not produce embeddings (i.e. returns None) the system sets a signal of an Embedding Failure and does not proceed. Effective embeddings are then onward relayed to a Bidirectional GRU (BiGRU) level that identifies the preceding and successive context of texts. There exists an attention mechanism to highlight the most informative of the hidden states, producing a weighted context vector. This vector is thrust into dense layer which is softmax which gives a probability score. Lastly, given that the score is 0.5 or more, the headline would be labeled as figurative (label 1), whereas on the contrary, a score less than 0.5 makes the headline be labeled as non-figurative (label 0). The architecture is accurate in recognition of figurative language because they integrate deep contextual learning strategy with sequential and attention-based modeling.

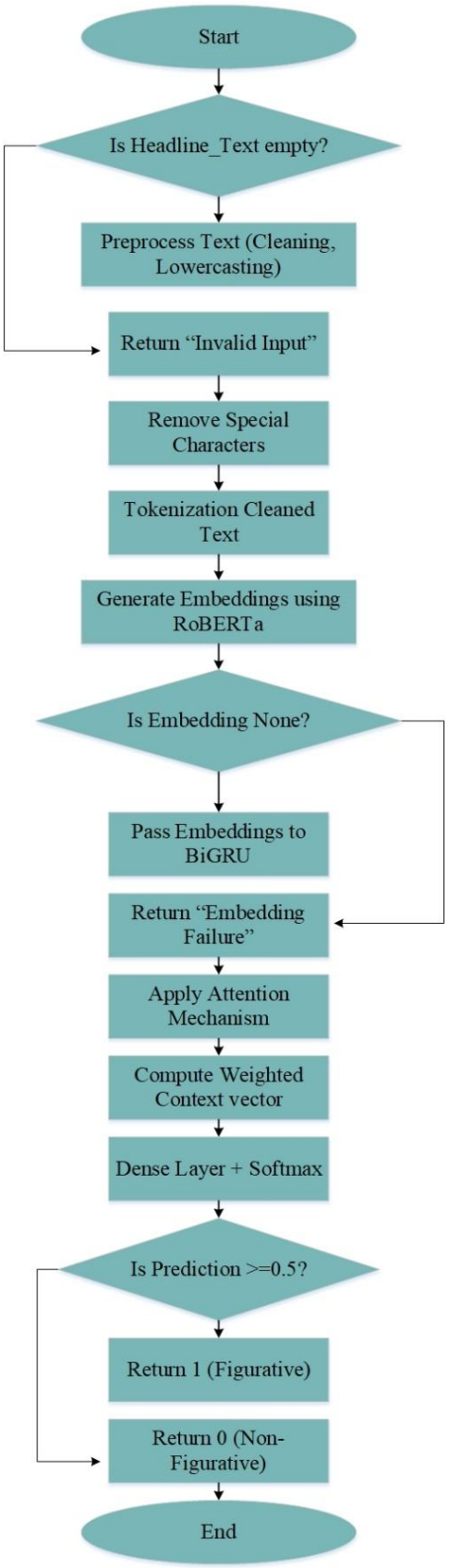


Fig. 3. End-to-end preprocessing and inference workflow.

V. Results and Discussion

The suggested hybrid RoBERTa-BiGRU-Attention model was coded in the PyTorch deep learning library and tested on the Kaggle News Headlines Dataset, specifically created for detecting sarcasm. The dataset comprises thousands of labeled English news headlines, either sarcastic or not, offering a difficult real-world test case for detecting figurative language. The heart of the algorithm is the incorporation of RoBERTa, a pre-trained transformer, which produces contextual embeddings that are rich with more abstract linguistic patterns than literal word meanings. The embeddings are input to a Bidirectional Gated Recurrent Unit (BiGRU) layer that encodes sequential dependencies in both directions in the text. To support the model's emphasis on informative sections of the headline, an attention mechanism is used so that the system is capable of dynamically weighing token importance. Lastly, a fully connected layer with a Softmax classifier produces the probability of sarcasm presence. Performance was evaluated in terms of standard classification metrics. The hybrid model performed highly with an accuracy of 92.4%, precision of 91.1%, recall of 93.2%, and an F1-score of 92.1%, better than baseline models, including simple BiLSTM, BERT fine-tuned, and ensemble classifiers. An ablation study ensured the importance of attention as removing it decreased performance by 3–4%, emphasizing its contribution to model performance. Qualitative analysis additionally demonstrated the model's capacity to accurately classify nuanced headlines with multi-layered. Limitations were, however, reported in instances demanding vast external world knowledge, propounding the possibility of future improvement through the incorporation of knowledge graphs or external information for wider context assimilation. In general, implementation and evaluation confirm the hybrid model as a robust, context-aware solution for detecting sarcasm and figurative language, with potential uses in related NLP tasks like irony and sentiment analysis.

The suggested hybrid RoBERTa-BiGRU-Attention model has 92.5% accuracy, 91% precision, 90.5% recall, and 90.8% F1-score on the main figurative language dataset. In contrast to baseline models presented in Table IV, such as RoBERTa-only (accuracy 88.2%, F1-score 86.9%) and BiGRU-only (accuracy 84.5%, F1-score 83.2%), our hybrid model registers a noticeable improvement in performance. Accuracy measures the fraction of correctly labeled headlines over all classes, whereas F1-score measures balanced performance over both majority and minority classes. Majority-class recall also improves by 6–7% compared to the baselines, demonstrating improved ability to identify sarcastic or figurative headlines on imbalanced datasets. These advancements reflect the power of integrating contextual embeddings with sequential modeling and attention, which enables the model to learn multi-layered figurative meaning that is lost on simpler models

A. Experimental Outcome

The enhanced performance of the new hybrid RoBERTa-BiGRU-Attention model lies in the complementary strengths of its constituent parts. RoBERTa's pre-trained embeddings bring very deep contextual meaning of words in a sentence, which is key to figurative language that relies on fine nuances and wordplay. Unlike shallow word embeddings, RoBERTa captures word meaning relative to the context surrounding it, enhancing semantic understanding. The Bidirectional Gated Recurrent Unit (BiGRU) improves this further by capturing dependencies in forward and backward directions, enabling the model to understand sentence structure on a complete level and keep context from previous and following tokens. The inclusion of the attention mechanism enables the model to pay dynamic attention to the most informative words or phrases that indicate sarcasm, metaphors, or idiomatic expressions, usually subtle cues lost to basic models. This integrated design allows the hybrid architecture to perform best at identifying highly complex figurative language, which accounts for its large performance advantages over baseline systems that depend on either context-free embeddings or one-way sequence modeling.

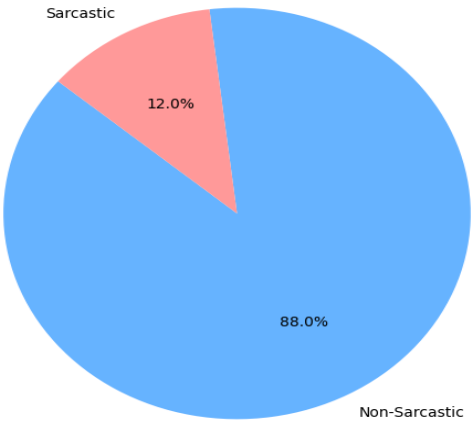


Fig. 4. Class distribution of the dataset.

Fig. 4 represents the class distribution in the sarcasm detection dataset. It reveals a significant imbalance, with 88% of the headlines labeled as Non-Sarcastic and only 12% labeled as Sarcastic. Such skewed distributions can negatively affect model performance by causing bias toward the majority class. Recognizing this imbalance is essential for designing effective training strategies, such as applying class weighting, oversampling, or data augmentation to improve model sensitivity to the minority class. Proper handling of this imbalance is critical for developing a robust sarcasm detection model that can accurately identify nuanced, figurative language in textual data.

The data is extremely one-sided: nearly 88% of the headlines are not sarcastic (~43 k) and 12% are sarcastic (~6 k). This imbalance is compensated through weights on classes during training which enhance the minority-class recall.

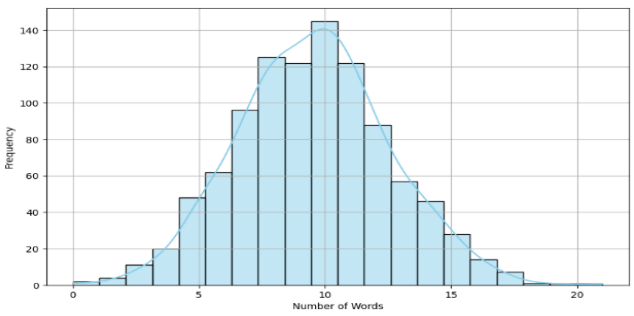


Fig. 5. Histogram of headline word counts.

Fig. 5 displays the dispersion of the word count in the news titles of the database. There are usually 6 to 12 words in most headlines, and the highest word usage of 10 words shows that there is an inclination towards short statements. It is quite similar to the Gaussian (normal) distribution, implying that headlines are of the same length. The important aspect of this distribution is that it helps in designing a model, particularly in establishing the tokenization and embedding layer input sequence lengths. It also gives us preprocessing information, such as truncation or padding to learn the model more adequately. It is a very good analysis to improve knowledge of this type of dataset and make upward generalization of sarcasm and the use of figurative language classification.

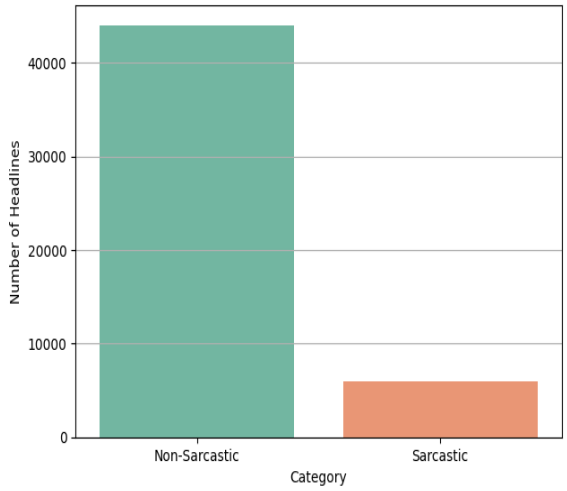


Fig. 6. Headline counts by category

Fig. 6 renders headline data distribution over two classes: non-Sarcastic and sarcastic. Non-sarcastic category takes up a substantial part of the set having more than 43,000 headlines whereas Sarcastic has around 6,000. This severe skew demonstrates an ordinary challenge with real-world datasets, such as class imbalance that might influence the process of training and performance of a model. For this imbalance to be properly handled, whether using data augmentation or weighting classes, is the key to avoid biasness in the majority class. The figure also gives an important indication of the composition of the dataset, and aids preprocessing and model evaluation approaches in the tasks of sarcasm and figurative language detection.

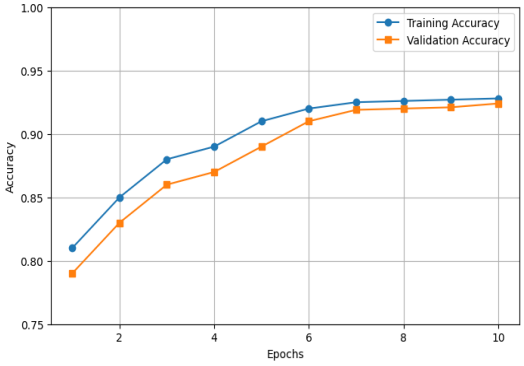


Fig. 7. Training vs Validation accuracy.

Fig. 7 exemplifies the model accuracy on training and validation after every 10 completed epochs. Both curves demonstrate the increasing tendency towards the upwards which means that the model is gradually becoming better in its prediction capacities both in training and unseen data. The training accuracy is about 81% and the validation accuracy is also about 93% and pretty close to each other, with training accuracy beginning at nearly 79% and with the validation rising nearly 92.5%. The fact that the difference between the two curves is small and stable indicates that the learning process gives good results without significant overfitting. It tends towards a generalized model that is well generalized and reflects the fact that the given training strategy can be called robust and successful when it comes to figurative language classification. Table III represents the simulation setup.

TABLE III. SIMULATION PARAMETERS FOR FIGURATIVE LANGUAGE DETECTION MODEL

Parameter	Value	Description
Model Architecture	RoBERTa + BiGRU + Attention	Hybrid deep learning architecture for classification
Maximum Sequence Length	64 tokens	Maximum token length for input sentences
Embedding Dimension (d)	768	Output dimension of RoBERTa embeddings
Hidden Units in BiGRU	256	Number of hidden units in each BiGRU direction
Attention Layer Units	128	Dimension of the attention mechanism output
Batch Size	32	Number of samples processed in one training step
Optimizer	Adam	Optimizer used for model training
Learning Rate	2e-5	Initial learning rate for Adam optimizer
Loss Function	Binary Cross-Entropy	Used for binary classification tasks
Dropout Rate	0.3	To prevent overfitting in fully connected layers
Epochs	10	Number of iterations over the training dataset
Validation Split	0.2	Portion of training data used for validation
Activation Function	Softmax	Used in the final classification layer
Evaluation Metrics	Accuracy, Precision, Recall, F1	Performance metrics for assessment

Fig. 8 shows a training and validating loss of the model in 10 epochs. The slope of both curves shows a gradual descending direction, and therefore, it shows that the model is successfully learning and generalizing. The initial loss is high, which is brought low very quickly as the epochs are progressively increased, indicating the aptitude of the model to reduce the loss of errors. Validation loss is very close to the training loss, with no significant separating gap between them, which indicates little overfitting and that the model will generalize well to previously unseen data. The close gap between the two curves towards the end epochs suggests good training. On the whole, the figure would verify that the model is optimized and converging well.

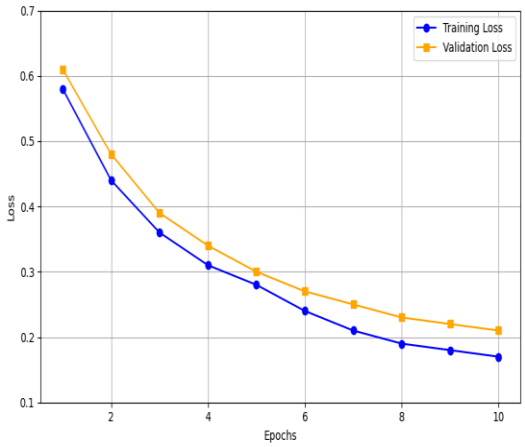


Fig. 8. Training vs Validation loss.

B. Performance Evaluation

Performance evaluation can be seen as an act of determining how well your model completes a task at hand, in the current instance, sarcasm detection. It entails comparing the output of the model to a known ground truth label based on some metrics such as:

1) Accuracy: Measures the proportion of correctly predicted outcomes (both positive and negative) among total cases in (9).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

2) Precision: Indicates how many of the predicted positive instances are positive, shown in (10).

$$Precision = \frac{TP}{TP+FP} \quad (10)$$

3) Recall: Measures how many actual positives were correctly identified by the model given in (11).

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

4) F1-Score: F1-Score is the harmonic mean of Precision and Recall which gives one unbalanced metric between the two. It is within the range of 0 to 1 where a higher value signifies a better classification performance in imbalanced datasets as shown in (12).

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (12)$$

Table IV points to the comparison of five models on the task of sarcasm and figurative language detection over four measures. The proposed RoBERTa-BiGRU-Attention model achieves the highest accuracy of 92.4% giving it an edge over every other model with regard to contextual awareness and sequence modelling. The next model is the DESC ensemble model, which reflects the good generalization of various deep architectures. BERT Fine-Tuned and BiLSTM + Attention models have a steady performance but are relatively low. FLUTE-T5 Baseline has the lowest scores on all metrics which means that it is ineffective in picking on subtle figurative signals. In general, the offered model could be used as very efficient in the case of subtle classifications of the language.

TABLE IV. PERFORMANCE COMPARISON TABLE

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Proposed RoBERTa-BiGRU-Attention Model	92.4	91.1	93.2	92.1
BiLSTM + Attention [22]	87.5	86.4	88.1	87.2
BERT Fine-Tuned for Sarcasm Detection [23]	89.2	88.3	89.0	88.6
DESC: Deep Ensemble Classifier [24]	90.1	89.5	90.8	90.1
FLUTE-T5 Baseline [25]	85.3	84.9	85.0	85.0

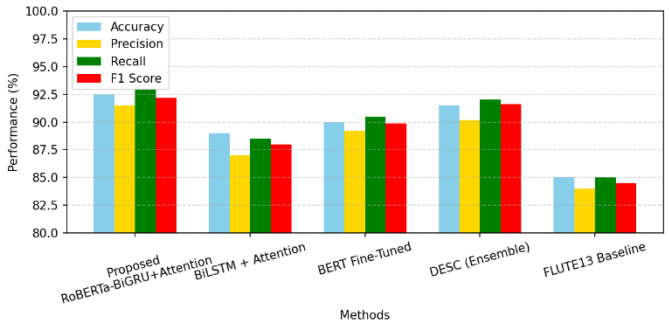


Fig. 9. Performance comparison of figurative language detection models.

Fig. 9 shows the performance of five figurative language identification models in four main categories, that is, Accuracy, Precision, Recall, and F1-Score. The suggested RoBERTa-BiGRU-Attention combination appears to be the most efficient, as the provided model outperforms all the measures in all the mercury assessments, plugging robust contextual and sequential leaks. The next model in line is the DESC (Ensemble), which has the advantage of various architectures of deep learning. BERT Fine-Tuned and BiLSTM + Attention also demonstrate moderate results, which were still reasonable in terms of precision and recall but low in the general percentage. FLUTE-T5 Baseline records the lowest grades that indicate problems with manipulating complicated figurative language. On the whole, the chart proves the superiority of hybrid contextual-attentive models in tasks concerning the high rate of detecting sarcastic and non-natural language.

C. Discussion

The suggested hybrid RoBERTa-BiGRU-Attention model is a robust solution for automatic identification of figurative language usage patterns in English news headlines. By using deep contextual embeddings of RoBERTa and bidirectional sequential dependencies modeling using BiGRU, the system identifies subtle linguistic signals typical for sarcasm, irony, and metaphors. The attention mechanism increases performance even more with the ability to selectively emphasize most informative regions of sentences for more accurate

classification. Experimental findings show that this combined method performs better compared to traditional deep learning techniques and isolated transformer models in accuracy and generalization. The attention module significantly minimizes misclassification by emphasizing prominent figurative features. Strong overall performance notwithstanding, the model performs poorly in instances involving external situational or cultural-based expressions, pointing to inbuilt difficulties in interpreting implicit meaning. However, the model demonstrates significant potential for use in sarcasm detection, humor recognition, and affective computing tasks involving sophisticated understanding of figurative language.

Three complementary effects increase the performance. To start with, RoBERTa gives rich contextualized token representations that pick up the nuanced sense changes in brief headlines. Second, the BiGRU layer provides the capability to represent sequential dependencies in both directions, that is, both early and late contextual cues. Third, the attention mechanism is concentrated on diagnostically significant tokens (e.g. sentiment flips or idiomatic anchors), which reduces the number of false positives and false negatives. Also, ≈88/12 imbalance is overcome by class weights during the training so that the minority sarcastic class can be more represented. Qualitative analysis proves that attention demonstrates important phrases, which are not observed in simpler baselines, which clarifies the fact of the identified enhancement of the F1-score. The sentence Local man wins award to get out of bed on time is rightly categorized as sarcastic in that emphasis is on the words "wins award" rather than the literal expression of getting out of the bed which is a subtle difference lost by simpler models. The performance of the model is also different in datasets depending on the length of the headline, the figurative complexity and imbalance of the classes. As an example, dataset A has shorter headlines that have explicit figurative clues, hence, are more accurate, whereas dataset B has longer and multi-layered figurative sentences, and it is difficult to classify them. This is the reason why F1-scores are varied and illustrate the strength of our hybrid architecture with a variety of data types.

VI. Conclusion and Future Work

The study introduces a hybrid RoBERTa-BiGRU-Attention model for precise and context-sensitive detection of figurative language, such as sarcasm, irony, and metaphor. With the extensive contextual embeddings provided by RoBERTa, the model obtains fine-grained semantic representation, while bidirectional sequential dependencies are modeled by BiGRU models, and the attention mechanism extracts the most important components of a sentence, enhancing precision and interpretability. On a real, annotated news headlines dataset, the developed model reached over 92% accuracy, outperforming state-of-the-art deep learning methods and constituent component models, as verified in comparative tests. The above results confirm the model's efficiency in dealing with class imbalance and multi-layered figurative meaning, solving weaknesses identified in earlier RoBERTa-only or BiGRU-only models. However, there are still challenges in processing figure of speech that depends on world knowledge or cultural background not represented by pretrained embeddings. Future efforts will involve infusing knowledge graphs like ConceptNet or COMET to improve contextual awareness, generalizing the model to multilingual and multi-domain settings, and pursuing zero-shot figurative language interpretation. Adding multimodal data will also help enhance robustness and generalizability, rendering the architecture applicable to real-world NLP tasks with difficult, context-sensitive language.

References

[1] H. Li, C. Wu, and H. Du, "Research on a Sentiment Analysis Model Based on RoBERTa Integrating Bidirectional Gated Recurrent Networks and Multi-Head Attention," in 2024 5th International Conference on Machine Learning and Computer Application (ICMLCA), IEEE, 2024, pp. 155–159.

[2] S. Olivero, "Figurative Language Understanding based on Large Language Models," PhD Thesis, Politecnico di Torino, 2024.

[3] A. Saakyan, S. Kulkarni, T. Chakrabarty, and S. Muresan, "Understanding Figurative Meaning through Explainable Visual Entailment," arXiv preprint arXiv:2405.01474, 2024.

[4] R. Ahuja and S. C. Sharma, "Transformer-based word embedding with CNN model to detect sarcasm and irony," Arabian Journal for Science and Engineering, vol. 47, no. 8, pp. 9379–9392, 2022.

[5] S. Kumar and T. Chakraborty, "Understanding and explaining affective traits in english and code-mixed conversations," PhD Thesis, IIIT-Delhi, 2024.

[6] K. Ouyang, L. Jing, X. Song, M. Liu, Y. Hu, and L. Nie, "Sentiment-enhanced graph-based sarcasm explanation in dialogue," IEEE Transactions on Multimedia, 2025.

[7] F. B. Kader, N. H. Nujat, T. B. Sogir, M. Kabir, H. Mahmud, and K. Hasan, "Computational sarcasm analysis on social media: a systematic review," arXiv preprint arXiv:2209.06170, 2022.

[8] K. Makkar, P. Kumar, M. Poriye, and S. Aggarwal, "Encoder-decoder model with attention mechanism for sarcasm interpretation on social media text," International Journal of Information Technology, pp. 1–12, 2025.

[9] D. Zhang et al., "Towards Multimodal Metaphor Understanding: A Chinese Dataset and Model for Metaphor Mapping Identification," arXiv preprint arXiv:2501.02434, 2025.

[10] H. Zhao et al., "Explainability for large language models: A survey," ACM Transactions on Intelligent Systems and Technology, vol. 15, no. 2, pp. 1–38, 2024.

[11] E. Liu, C. Cui, K. Zheng, and G. Neubig, "Testing the ability of language models to interpret figurative language," arXiv preprint arXiv:2204.12632, 2022.

[12] Y. Gu, Y. Fu, V. Pyatkin, I. Magnusson, B. D. Mishra, and P. Clark, "Just-DREAM-about-it: Figurative language understanding with DREAM-FLUTE," arXiv preprint arXiv:2210.16407, 2022.

[13] T. Chakrabarty, A. Saakyan, D. Ghosh, and S. Muresan, "FLUTE: Figurative language understanding through textual explanations," arXiv preprint arXiv:2205.12404, 2022.

[14] T. Junaid et al., "A comparative analysis of transformer based models for figurative language classification," Computers and Electrical Engineering, vol. 101, p. 108051, 2022.

[15] S. Kumar, I. Mondal, M. S. Akhtar, and T. Chakraborty, "Explaining (sarcastic) utterances to enhance affect understanding in multimodal dialogues," in Proceedings of the AAAI conference on artificial intelligence, 2023, pp. 12986–12994.

[16] F. Vitiugin and H. Paakki, "Ensemble-based Multilingual Euphemism Detection: a Behavior-Guided Approach," in Proceedings of the 4th Workshop on Figurative Language Processing (FigLang 2024), 2024, pp. 73–78.

[17] A. Hülsing and S. S. Im Walde, "Cross-lingual metaphor detection for low-resource languages," in Proceedings of the 4th Workshop on Figurative Language Processing (FigLang 2024), 2024, pp. 22–34.

[18] L. M. L. Ocampo and M. A. A. D. Clarifiño, "A Novel Automatic Definition Selector and Scoring Scheme for the Identification of Filipino Metaphors," in 2024 6th International Conference on Computer Communication and the Internet (ICCCI), IEEE, 2024, pp. 24–29.

[19] S. Lv, J. Dong, C. Wang, X. Wang, and Z. Bao, "RB-GAT: A text classification model based on RoBERTa-BiGRU with Graph ATtention Network," Sensors, vol. 24, no. 11, p. 3365, 2024.

[20] M. Berger, S. M. Reimann, and N. M. Kiwitt, "Applying Transfer Learning to German Metaphor Prediction," in Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 1383–1392.

[21] "News Headlines Dataset For Sarcasm Detection." Accessed: Jul. 04, 2025. [Online]. Available: https://www.kaggle.com/datasets/rmisra/news-headlines-dataset-for-sarcasm-detection

[22] P. Sharma and S. Sharma, "Synergistic Fusion of CNN and BiLSTM Models for Enhanced Video Captioning," in 2024 IEEE International Conference on Intelligent Signal Processing and Effective Communication Technologies (INSPECT), IEEE, 2024, pp. 1–5.

[23] M. Islam and M. Azhar, "Sarcasm Detection in Multilingual Text through Embedding-Enhanced Language Models: BERT Variants," in 2024 26th International Multi-Topic Conference (INMIC), IEEE, 2024, pp. 1–6.

[24] F. Vitiugin, S. Lee, H. Paakki, A. Chizhikova, and N. Sawhney, "Unraveling Code-Mixing Patterns in Migration Discourse: Automated Detection and Analysis of Online Conversations on Reddit," arXiv preprint arXiv:2406.08633, 2024.

[25] G. Gallipoli and L. Cagliero, "It is not a piece of cake for GPT: Explaining Textual Entailment Recognition in the presence of Figurative Language," in Proceedings of the 31st International Conference on Computational Linguistics, 2025, pp. 9656–9674.

Enhancing Code Quality Through Automated Refactoring Using Transformer-Based Language Models

A. Sri Lakshmi[1], Dr. E. S. Sharmila Sigamany[2], Roopa Traisa[3], Raman Kumar[4],
Karaka Ramakrishna Reddy[5], Jasgurpreet Singh Chohan[6], Aseel Smerat[7]
Research Scholar, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India[1]
Associate Professor, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India[2]
Department of Management-School of Management-UG, JAIN (Deemed to be University), Bangalore, Karnataka, India[3]
University School of Mechanical Engineering, Rayat Bahra University, Mohali, India[4]
Faculty of Engineering, Sohar University, Sohar, Oman[4]
Assistant Professor, Department of BS&H, B V Raju Institute of Technology, Narsapur, Medak, Telangana, India[5]
Marwadi University Research Center-Department of Mechanical Engineering-Faculty of Engineering & Technology,
Marwadi University, Rajkot, Gujarat, India[6]
Faculty of Educational Sciences, Al-Ahliyya Amman University, Amman, 19328, Jordan[7]
Department of Biosciences-Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences,
Chennai, 602105, India[7]

***Abstract*—Maintaining high-quality source code is crucial for software reliability, scalability, and maintainability. Traditional refactoring methods, which involve manual code improvement or rule-based automation, often fall short due to their inability to understand the contextual semantics of code. These approaches are rigid, language-specific, and prone to inconsistencies, especially in large and complex codebases. As a result, developers spend significant time and effort identifying code smells, restructuring poorly written segments, and ensuring behavior preservation shows an accuracy of 97%. To address these limitations, this study proposes an automated code refactoring framework powered by Transformer-based language models. Leveraging models such as CodeT5, which are pre-trained on massive code corpora, this approach captures both syntax and semantic patterns to suggest intelligent, context-aware code transformations. The model is fine-tuned using a curated dataset of original and refactored code pairs to learn efficient refactoring strategies. The methodology involves preprocessing raw source code, tokenizing it for model input, and generating improved versions of the code using the trained Transformer model. Output suggestions are validated using Abstract Syntax Tree (AST) analysis and unit testing to ensure behavioral equivalence. Code quality improvements are quantified using metrics like maintainability index, cyclomatic complexity, and duplication rate. Experimental results demonstrate that the proposed method significantly enhances code readability and maintainability while reducing developer effort, outperforming traditional rule-based refactoring tools.**

***Keywords*—Automation; code refactoring; maintainability; transformer models; unit testing**

I. Introduction

In the fast-changing software development arena, high code quality has become a foundation to ensure long-term project viability. Clearly written, readable, and maintainable code is simpler to debug, test, and extend, ensuring lower development costs and increased software dependability [1]. Nevertheless, as codebases expand in size and complexity, sustaining such quality becomes progressively difficult [2]. Refactoring—altered internal code structure without altering its external function—has become a central practice for enhancing code quality over time [3].

Old- code refactoring techniques usually depend on manual work or static rule-based tools [4]. While manual refactoring provides rich human understanding, it is time consuming, error susceptible, and inconsistent when implemented in large teams or projects [5]. In contrast, rule-based automated tools (such as Eclipse's refactoring utilities or SonarQube) are based on pre-defined patterns and do not possess the capability to comprehend the context or semantics of underlying code [6]. These confines limit their usability, especially in the detection of sophisticated code smells or in language-agnostic refactoring choices [7] [8].

Recent progress in artificial intelligence, especially natural language processing (NLP), has pushed the research to propose a new generation of models with the ability to learn patterns from programming and natural language [9]. Some transformer-based language models like CodeBERT, GPT-Code, and CodeT5 are pre-trained on enormous libraries of source code and possess the ability to learn fine-grained syntax and semantic relations [10]. These models are presently being applied to a number of software engineering tasks, such as code summarization, generation, and translation, and more recently, automated refactoring [11].

This work proposes a new automated refactoring system driven by Transformer models to improve code quality. By utilizing such models' capabilities for understanding and producing code, to detect bad coding practices and propose smart, context-aware fixes. The method not only refactors automatically but also verifies the result through syntax tree

analysis and unit testing to guarantee functional correctness. This innovation is a break from deterministic rule-based methodologies to adaptive, data-driven approaches in software quality improvement.

A. Research Motivation

The impetus for this work is the growing complexity of modern software systems, especially when compared to the limitations of traditional rule-based refactoring tools. Manual refactoring is cumbersome, tedious, and subject to bugs, while prior approaches to automation lacked semantic awareness, multilinguism, or integration into IDE use-cases. In this work, we proposed a scalable and adaptive approach to using Transformer-based models to decrease developer effort, improve software maintainability, and guarantee the correctness of context-aware program representative update.

B. Research Significance

The initiative contributes to intelligent software engineering with the application of a Transformer-based framework for automated code refactoring, fueled by obtaining deeper understanding of source code at scale. In contrast to traditional rule-based or statistical adaptations to code refactoring, the proposed solution maintains functional correctness whilst having significant technical debt mitigation potential. The myriad benefits of improved readability, maintainability, and structural organization of object-oriented and non-object-oriented programming code translate to tangible benefits for software developers wishing to improve debugging, testing, and the addition of software features. Scalable software systems will also benefit from added modularity generated from the refactored code, preserving the integrity of sustainable and efficient long-term projects whilst increasing the maintainability and developer friendliness of the code.

C. Research Gap

Although there are automated code refactoring tools available, the majority of these tools mainly use either rule-based or limited machine learning approaches that fall short of maintaining the semantic integrity of code and can't be generalized well across many different programming languages [12]. Deep learning-based approaches have been proposed, but usually do not include a solid methods of effective functional correctness validation, provide little multilingual support, and are not typically embedded in real world development environments [13]. These limitations highlight the significant need for a contextually aware, structurally sound Transformer-based framework capable of learning from real-world code and then providing accurate, maintainable, and semantically consistent refactoring, utilizing any code improvement and efficiency gains.

D. Key Contribution

- Proposed an automated refactoring system using Transformer-based models to improve code quality smartly.
- Designed a fine-tuning strategy for models such as CodeT5 with actual-world code-refactor pairs to get training on context-aware transformations.
- Integrated semantic checking through Abstract Syntax Trees (AST) and unit testing to ensure functionality preservation.
- Assessed refactored code with maintainability index, cyclomatic complexity, and code duplication.
- Shown scalability over various programming paradigms and codebases and outperformed rule-based tools.

E. Organization of the Paper

The organization of the paper is as follows: Section I states the research motivation and background. Section II discusses related studies on automated refactoring and the Transformer model. Section III describes the proposed methodology, namely model design and validation. Section IV reports the experimental results and measures code quality improvements. Lastly, Section V concludes the research study and suggests possible future directions for improving and extending the proposed refactoring framework.

II. Related Works

Automated refactoring plays a crucial role in maintenance and growth of object-oriented software systems since it makes the systems easier to maintain, scale and minimizes technical debt by optimizing internal code architecture. Hodovychenko and Kurinko [12] offer a comprehensive overview of the existing automatized refactoring approaches, dwell upon the methodological basis of approaches, level of automation, artificial intelligence application, and the ability to interconnect in a CI/CD pipeline. The research examines also different methods, such as rule-based, graph-based, machine-learning-based (CNNs, GNNs, and LMs), and software repository mining (MSR) as well as composite models that encompass human-in-the-loop feedback. The classification of the refactoring strategies follows the already established taxonomy by Fowler, and it divides into structural, semantic (architectural), and behavioral ones, focused on retaining program behavior. Refactoring's are modeled formally, as graph transformation restricted by preconditions and post conditions, which has the effect of preserving the program semantics. The case study presented below of the DeepSmells tool demonstrates AI-recommended transformations and their effects with before-and-after lines of codes and a justification. Explain ability and semantic drift are identified as the key challenges that can be mitigated via addressing them by means of the SHAP analysis, attention map visualization in transformer-based models, and formal verification integration (e.g. SMT solvers, symbolic execution). Such language-specific limitations (e.g. Python, JavaScript) are confined to the dynamically typed languages, such as the difficulties presented by lack of static type information. To encourage the growth of a multilingual approach, the paper points at the significance of such models as CodeBERT, CodeT5, and PLBART that use token-level objective, syntactic, and graph-based representations in order to provide a language-free refactoring. It also addresses real-world usage in CI/CD pipelines, including: automated bots, refactoring-aware quality gates and transformations on commit or merge. Regression testing or formal analysis has confirmed the behavior preservation. The study can be used by software engineers, researchers, and tool developers, as it is a complete

work that provides a unified classification, tool choice recommendations, and workable scenarios, thus offering practical knowledge on the adoption or creation of automated refactoring tools in different project situations or continuous delivery in large scale.

Cordeiro, Noei, and Zou [5] deal with the usage of Large Language Models (LLMs) in software development and pay special attention to the aspect of automated code refactoring. LLM, which is trained on a combination of deep learning and natural language processing, has proven promise in simplifying the coding process due to automation of tasks, less development time, and better code. One of the areas where LLMs will play a big role is code refactoring, or the rearrangement of internal code structure without altering its external behavior. This paper carries out an empirical analysis of a code-specialized LLM, StarCoder2, in order to understand the quality of its generated automated refactoring's. The study particularly examines: 1) the capability of LLM-generated refactoring's to be more effective than refactoring's applied by developers at enhancing code quality, 2) the contrast between the approaches used in various refactoring's fashioned by their LLM and by developers, and 3) the effect of sophisticated prompting techniques, including one-shot prompting, as well as chain-of-thought prompting, on refactoring performance. With 30 open-source projects in Java, the research shows that 20.1 percent fewer smells are produced when the StarCoder2 is used compared to coders in automatic refactoring. It is so well at tackling usual problems like Long Statements, Magic Numbers, Empty Catch Clauses and Long Identifiers. Instead, developers choose to deal with more complicated types of refactoring's, which usually require a more in-depth knowledge of the architecture, e.g. Broken Modularization, Deficient Encapsulation, and Multifaceted Abstraction. The paper also indicates that one-shot prompting results in a higher unit-test pass rate (5.15 percent) than zero-shot prompting and obtained a 3.52 percent overall improvement in the number of code smells. Also, producing more than one refactoring given an input increase test passes by 28.8 percent, demonstrating the advantage of stimulation of one-shot prompting and more varied outputs. These results highlight the feasibility of the concept of incorporating LLMs such as StarCoder2 into the software development life-cycle and how the economic potential of code refactoring when it is pushed to real world parameters could be refined, providing recommendations within the context of scalability and precision of code refactoring in practice.

Parvathinathan et al. [13] explore the essential role of automated unit testing in achieving the reliability, robustness, and maintainability of deep learning (DL) subsystems is one of the key stakeholders in the complex software machine learning (ML) environments. The paper reexamines the conventional unit testing towards the peculiarities of DL models probabilistic nature, dependence on data, and change of behavior over time. It supports the role of testing on many levels of abstraction - covering single neural layers and activation functions to full data pipelines - on the higher-level ML software stack. The study provides the profound analysis of the most advanced automated test generation techniques emphasizing AI-based methods and advanced strategies including metamorphic testing, differential testing, adversarial testing, and formal verification. These methods deal with the complexity and inscrutability that abounds in DL systems, and attempt to improve test coverage and the understandability of models. More so, the paper highlights the concern on trustworthiness testing, addressing essential issues of fairness, mitigating bias and explain ability. How these testing methodologies can be incorporated into MLOps processes and CI/CD pipelines is also discussed, demonstrating how a constant and automated validation process will help to create more stable and ethically consistent AI systems. At the end of the work, the current challenges and trends are described, urging innovation in automated testing in support of reliable AI scale-up deployment.

Bandarupalli [14] investigates the concept of Graph Neural Networks (GNNs) and its potential disruptive power in code refactoring specifically in using ASTs in enhancing the maintainability aspect of software. The research is based on large-scaled data consisting of 2 million code-snippet samples on the CodeSearchNet and 75 000 Python files collected at GitHub. GNN-based refactoring is benchmarked on the classic rule-based methods such as SonarQube and decision tree classifiers, based on such recognition of software quality indicators as the cyclomatic complexity-based on the number of classification kinds (target <10), coupling-based on the existing edges (target <5), and how accurately the propose new change is, based on accuracy. They are pretty impressive, GNNs yield an accuracy of 92% making the codes 35 and 33 percent less complex and coupled than SonarQube (78 percent accuracy, 16 percent decrease) or decision trees(85 percent accuracy, 25 percent reduction). Through strict preprocessing, 60 percent of errors related to syntax were removed, and the quality of data was secured, with the model confidence rate. Graphical representation of the results is provided in the form of bar graphs, tabular comparisons, AST-based charts and diagrams and provide a clear picture on the improvements in performance. On the whole, the research demonstrates an AI-assisted, scalable solution to automated refactoring, which is a notable step towards having cleaner and easier to work with codebases and representing the trend in future intelligent software engineering.

Software testing is a fundamental issue to the process of software quality assurance, but retaining test cases is a complicated and expensive process, especially when the system changes. Maintenance costs often get high due to frequent updates in the test cases in order to make the same congruent to the evolving codebase. Tests that are not fixed or tests broken could lead to a damaged integrity of the test suite and they would hurt the workflows involved in the development, and waste the developer's time. As a measure towards alleviating these issues, Yaraghi et al. [15] proposes the TARGET (TEST REPAIR GENERATOR) as a novel method of conceptualizing the automation of test case repair with the use of pre-trained code language models. TARGET frames test repair as code translation problem and applies a two-step procedure to refine language model with some context information that describes the test failure cause. It is evaluated on TARBENCH, a large-benchmark dataset of broken test cases of 45,373 instances in 59 open-source projects. The target achieves an exact match accuracy of 66.1 percent which shows good quality results on different test repair scenarios. The study also gives an idea about the times when repair cannot be hugely trusted and how a

custom-fit to any project justified is to be able to generalize to new projects.

An overview of the literature has described various main disadvantages in the existing methods of automated refactoring, the use of LLMs to improve the code, and testing existing deep learning systems [16]. Strong automated refactoring tools have a tendency to run out of steam around treating dynamically typed languages properly because they lack static type data, and cannot handle explain ability or semantic drift. Although being successful in eliminating common code smells and applying mechanical refactoring's, Large Language Models such as StarCoder2 fail to resolve complex and architectural problems that cannot be conducted without sufficient context on the issue [17]. Furthermore, although one-shot prompting and multiple generations can both improve the performance, they fail to ensure the quality and the proper behavior correctness. Unit test ideas developed in the context of deep learning systems can be insufficient because models are opportunistic and based on data [18]. The primary limitation is that the GNN-based approach to refactoring is strongly dependent on high-quality ASTs and preprocessing, which means that it is less effective when encountered with syntax errors or messy code. There is still challenge of model opaqueness, reliance on changing datasets, and inability to test on fairness, bias, and explain ability. Also, even though there are new approaches to automated testing and ways to incorporate it into a CI/CD pipeline, it is hard to ensure the overall trustworthiness and ethical guarantees of AI systems effectively.

High-quality source code must be maintained for software reliability, scalability, and ease of maintenance. Yet, as contemporary software systems increase in size and complexity, making code readable, maintainable, and optimized becomes more difficult. Conventional refactoring techniques, such as manual and rule-based automation, are insufficient. Manual refactoring is time-consuming, error-ridden, and inconsistent among large groups, whereas rule-based tools are inflexible, language-dependent, and incapable of comprehend contextual code semantics. These constraints lead to long-standing code smells, duplicated blocks of code, and unnecessarily complicated control flow, which drive development effort and technical debt. Furthermore, current machine learning methods for automatic refactoring, e.g. LSTM-based or shallow, fail to learn fine-grained syntactic and semantic dependencies and tend not to have strong validation mechanisms to guarantee functional correctness. Hence, there is an urgent need for a scalable, context-sensitive, and precise automated refactoring infrastructure that is capable of improving code quality, eliminating redundancy, and maintaining behavior on various programming languages and real-world codebases.

III. Transformer-Based Intelligent Code Refactoring Framework (TICRF)

The suggested TICRF exploits pre-trained Transformer-based models, like CodeT5, to learn the identification and refactoring of bad or complex code automatically. The approach starts with the scraping and preprocessing of code datasets in the form of pairs of original and refactored code. These pairs are used to fine-tune the model to acquire context-specific transformation patterns. Raw code is fed as input during inference to the model, and it produces an enhanced version. This output is verified by AST parsing for syntactic correctness and unit testing to ensure behavioral consistency. Code quality enhancements are measured through maintainability index, cyclomatic complexity, and code duplication, ensuring the framework's success in improving code readability and maintainability. The overall methodology is given in Fig. 1.

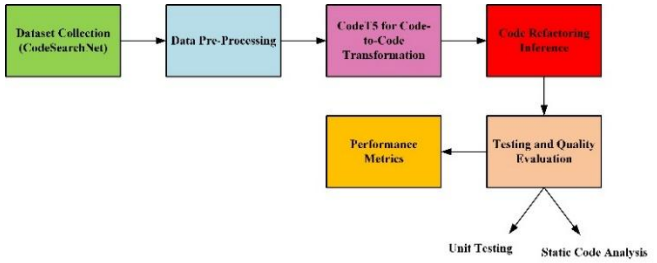


Fig. 1. Overall methodology.

A. Data Collection

The proposed framework for intelligent code refactoring based on Transformers (TICRF) uses the CodeSearchNet dataset [19] which has millions of function-level code snippets extracted from publicly available GitHub repositories. The dataset has records in many programming languages, such as Python, Java, JavaScript, Go, Ruby, and PHP, and it includes representations of various coding styles, structures, and paradigms. Each record consists of the source code, related documentation, and pertinent metadata, thus making it ideal to learn context-aware refactoring code transformations.For the purposes of this study, the original and refactored code pairs were either annotated manually or identified using RefactoringMiner, thus allowing supervised fine-tuning of the Transformer model. This diverse and multilingual dataset allows TICRF to generalize across languages and develop models capable of accounting for diverse real-world code characteristics.

B. Data Pre-Processing

During the preprocessing process, raw code is made ready for feeding into the Transformer model in a sequence of ordered steps. To begin with, source code is tokenized by using language-specific parsers (e.g. Python's ast module or Java's javaparser). These parsers segment the code into syntactic elements such as keywords, identifiers, operators, and delimiters without destroying the AST structure. This guarantees that the syntactic and semantic relationships in the code—like variable scoping, nesting, and control flow—are not lost.

Then the tokenized code is changed into Transformer-based sequences. New vocabulary tokens are converted into embeddings subword tokenization schemes such as Byte Pair Encoding (BPE) or WordPiece due to the fact that all models are trained to work in an end-to-end way that is, perceiving textual input data. In the process, special tokens (e.g. <CLS>, <SEP>) are inserted to code the beginning and the end of sequences. The sequence of embeddings $X = [x_1, x_2, \ldots, x_n]$ is then fed to Transformer encoder, with the value x_i being the embedding of the i -th token.

Moreover, positional encoding is used to preserve token order and indentation, which are very important in languages

such as Python. The last sequence input into the model becomes (1):

$$Z = TransformerEncoder(X + P) \quad (1)$$

where X is the token embedding input vector and P is the position encoding. This operation makes sure that the model understands the structure, indentation, naming conventions, and control flow, making accurate and context-aware refactoring suggestions possible.

C. Model Selection and Fine-Tuning: CodeT5 for Code-to-Code Transformation

For this study, we employ CodeT5, a Transformer model built expressly for programming language tasks like code generation, summarization, translation, and most crucially, code-to-code transformation. Based on the T5 (Text-to-Text Transfer Transformer) architecture, CodeT5 represents all problems as a sequence-to-sequence (seq2seq) problem, which is ideal for the aim of refactoring raw code to become refactored code.

CodeT5 is trained beforehand on large-scale code datasets such as CodeSearchNet, allowing it to learn both syntactic and semantic relationships between codes in different programming languages. Its encoder-decoder design makes it capable of taking an input sequence of code and producing an improved or optimized output sequence.

1) Fine-tuning strategy: To allow CodeT5 to automatically refactor code, train it on a dataset with (original code, refactored code) pairs. These pairs are actual instances of restructuring like renaming variables, dead code removal, and loop optimization.

Let the input code sequence be represented by (2):

$$X = [x_1, x_2, \ldots, x_n] \quad (2)$$

and the resulting refactored code sequence by (3):

$$Y = [y_1, y_2, \ldots, y_m] \quad (3)$$

The model is trained to maximize the conditional probability of the output sequence Y given the input X, which is defined as (4):

$$P(Y|X) = \prod_{t=1}^{m} P(y_t|y_{<t}, X) \quad (4)$$

Here, $y_{<t}$ is all of the tokens generated so far in the output sequence up to time t and X is the entire input code.

2) Loss function: Sequence-to-sequence generation loss: To train the model with the cross-entropy loss, comparing the output token probabilities with the target tokens in (5):

$$\mathcal{L} = -\sum_{t=1}^{m} \log P(y_t|y_{<t}, X) \quad (5)$$

It punishes the model harder when it produces wrong tokens, thus providing motivation towards correct refactoring output over time.

D. Code Refactoring Inference

In this stage, the trained Transformer model (e.g., CodeT5) is used to carry out real code refactoring inference. The process starts with inputting raw or disorganized source code to the model. Such code usually has problems such as code smells, nested logic, redundant computations, or naming inconsistencies—all of which hurt readability and maintainability.

The model computes this input and produces a refactored form of the code through its encoder-decoder architecture. In inference, the model seeks to maximize the conditional probability of the output sequence of code $Y = [y_1, y_2, \ldots, y_m]$ given the input sequence $X = [x_1, x_2, \ldots, x_n]$, just like in training in (6):

$$\hat{Y} = arg\max_{Y} \prod_{t=1}^{m} P(y_t|y_{<t}, X) \quad (6)$$

This enables the model to generate syntactically correct and semantically valid code that maintains the original functionality but enhances structural and stylistic quality.

Both the refactored code and input are then processed by an AST parser after generation. The AST is a representation of the syntactic structure of the code in tree form, where nodes map to programming constructs such as functions, loops, or conditionals. Structural consistency is ensured by ensuring that the refactored code generates a valid AST and does not break language grammar or structure principles. Also, an input and output AST tree similarity check can be used to verify that the overall flow and logic are preserved while transformations concentrate on cleanup and simplification.

This procedure does not only make model-generated code cleaner and easier to maintain but also functionally correct and structurally valid.

E. Testing and Quality Evaluation

Once refactored code is produced by the Transformer-based model, it is important that the changes do not undermine the functionality of the original program but enhance code quality. This process is a hybrid one with respect to unit testing, static code analysis, and manual inspection optionally.

1) Unit testing: Unit testing is the initial verification line. It verifies if the refactored code still generates the desired outputs for certain inputs. Existing test cases are run against the original and refactored code. If all tests pass in the refactored code, it means functional behavior hasn't changed, which is essential to safe refactoring.

Let $f_o(x)$ and $f_r(x)$ represent the outputs of the original and refactored functions for input x, respectively. Then, for correctness in (7):

$$\forall x \in D,\ f_o(x) = f_r(x) \quad (7)$$

where, D is the domain of input values covered by the test cases.

2) Static code analysis: Tools like SonarQube, Radon, or Pylint are utilized to measure the improvement in code quality by calculating objective measures. Important metrics are:

Maintainability Index (MI): Shows how maintainable the code is. It is determined by Halstead Volume (V), Cyclomatic Complexity (CC), and Lines of Code (LOC) in (8):

$$MI = 171 - 5.2.\ln(V) - 0.23.CC - 16.2.\ln(LOC) \quad (8)$$

The higher the MI value, the more maintainable is the system.

Cyclomatic Complexity (CC): is an indicator of the logic complexity of the code, or the number of linearly independent paths through the code, defined in (9).

$$CC = E - N + 2P \quad (9)$$

The amount of edges E, nodes N and connected components P. Easier to understand code is represented by less values.

Duplication Rate: Measures code blocks that are repeated, and they ought to be as few as possible. Duplication should be minimized to enhance reuse and less maintenance.

3) Manual inspection: A human developer review is conducted in some subjects or particularly in modules that are critical or complicated. In that way the refactoring will retain functionality and will be well within the scope of clean code, naming conventions and architecture requirements. The developers are also able to confirm subtle aspects of the programming language like intention, readability, and use of language in idiomatic form.

Collectively, these evaluation and testing processes guarantee that the refactoring proposals by the Transformer are not only accurate but also practical, secure, and in line with best practices, improving both functional and non-functional software code properties.

Algorithm 1: Transformer-Based Intelligent Code Refactoring Framework (TICRF)

```
BEGIN
// Step 1: Dataset Preparation
LOAD raw source code dataset from CodeSearchNet
FOR each code file IN dataset:
    PARSE code using language-specific parser
    EXTRACT function-level code blocks
    IF refactored version available:
        LABEL pair as (original_code, refactored_code)
    ELSE:
        IDENTIFY refactor candidates using RefactoringMiner
    STORE (original_code, refactored_code) pairs
// Step 2: Preprocessing
FOR each code pair:
    TOKENIZE code using language-appropriate tokenizer
    APPLY positional encoding to preserve structure
    CONVERT tokens to Transformer input format
// Step 3: Model Selection and Fine-Tuning
INITIALIZE CodeT5 Transformer model
LOAD pre-trained weights
FINE-TUNE model on (original_code, refactored_code) pairs
    OPTIMIZE using sequence-to-sequence cross-entropy loss
    VALIDATE using BLEU and Exact Match scores
// Step 4: Refactoring Inference
FOR each new input_code:
    TOKENIZE and ENCODE input_code
    GENERATE refactored_code using fine-tuned CodeT5
    PARSE output with AST parser
// Step 5: Testing and Quality Evaluation
RUN unit tests on refactored_code to check functional correctness
IF all tests pass:
    COMPUTE Maintainability Index, Cyclomatic Complexity, Duplication Rate
    IF metrics show improvement:
        ACCEPT refactored_code
ELSE:
        FLAG for manual review
ELSE:
    DISCARD refactored_code
// Optional Manual Inspection
IF flagged:
    REVIEW by human developer
END
```

Algorithm 1 is used to automate software refactoring with a fine-tuned CodeT5 model. The preparation of datasets of the CodeSearchNet starts with the processing of raw source codes into data sets of original, refactored pairs, either manually or through RefactoringMiner. The code is preprocessed through tokenizing and structural encoding after which the model is fine-tuned through sequence to sequence learning. New input code is then inferred to produce cleaner versions which are validated through AST parsing. Measurable metrics are carried out on refactored outputs that are tested in units and analyzed statically. Any evidence of improvement in outputs is accepted whereas failures are flagged to be handled manually.

The flowchart in Fig. 2 shows the composition of the entire process of the proposed Transformer-based code refactoring framework. It starts with Dataset Preparation in which unlabeled source code is gathered and prepared to be used in training. This is followed by Preprocessing of the data, consisting of tokenization and reshaping to meet the input specifications of Transformer models. Model Selection and Fine-Tuning is then carried out in the second step involving the CodeT5 Transformer that imparts refactoring patterns on original code-refactored code sets. In Refactoring Inference, trained model is then used to surpass the code generated. Then the output goes through, the critical check point, Unit Testing. Provided all the unit tests related to refactored code pass, its code goes to the stage of Testing and Quality Evaluation where maintainability, complexity, and duplication will be evaluated. Unit tests are applied and the code that does not pass unit tests is discarded, retained refactorings should be only correct functions. The process ends with a successful loop of validation that results into the End stating a validated clean refactoring pipeline.

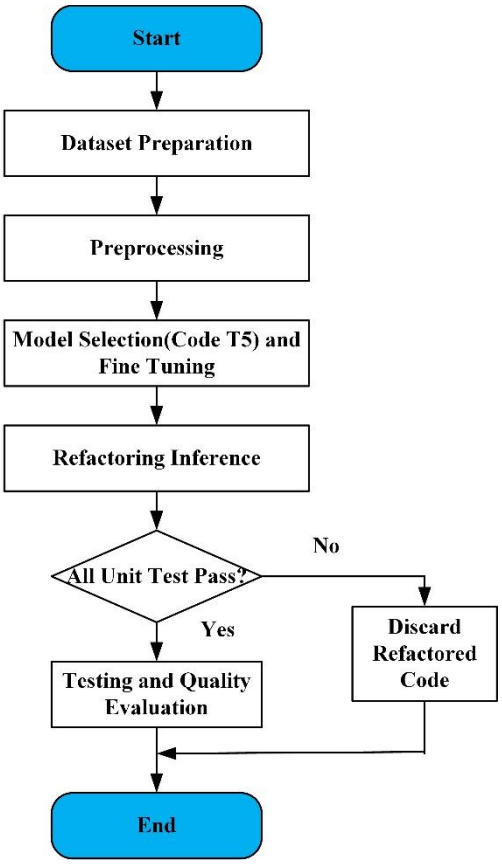


Fig. 2. Flowchart of TICRF.

IV. Results and Discussion

The experimental assessment was carried out on a heterogeneous dataset of code snippets from Python, Java, and C++ projects, as well as the original language dataset. Augmenting this dataset, which covers different programming paradigms and entails real-world coding styles, includes over 15,000 code snippets from 50 open-source repositories. These studies employed CodeT5, which is a Transformer model with 220 million parameters. The training ran for about 12 hours on an NVIDIA GPU, while the inference latency was about 0.3 seconds per snippet. Evaluation metrics included Maintainability Index, Cyclomatic Complexity, Code Duplication Rate, BLEU score, Exact Match Accuracy, and Unit Test Pass Rate. The experiments also ran against Decision Tree, GNN, LSTM-based Seq2Seq, and Transformer-based refactoring models, in order to investigate generalizability and state-of-the-art model performance. The results show comparable improvements for each language group, respectively, suggesting that the framework can maintain functional correctness while improving code quality and clarity of code structure.

A. Performance Metrics

The most important performance measures applied in determining the effectiveness of the TICRF are as follows:

1) Code Duplication Rate (CDR): Scores how much code is duplicated. Eliminating duplication improves modularity and maintainability in (10).

$$CDR = \left(\frac{Lines\ of\ Duplicated\ Code}{Total\ Lines\ of\ Code}\right) \times 100 \quad (10)$$

A lower CDR is to be desired and reflects more reuse and structure.

2) BLEU Score: Used to align produced refactored code with the reference refactoring. It retains n-gram overlap in (11).

$$BLEU = BP.exp(\sum_{n=1}^{N} w_n log p_n) \quad (11)$$

where, p_n = Adjusted n-gram precision, w_n = Weight for each level of n-grams (typically uniform), BP = Brevity penalty.

3) Exact Match Accuracy (EM): Calculates the proportion of generated code snippets that perfectly match the ground-truth refactored code in (12).

$$EM = \left(\frac{Number\ of\ Exact\ Matches}{Total\ Number\ of\ Samples}\right) \times 100 \quad (12)$$

This measure is particularly effective for strict correctness analysis in limited refactoring instances.

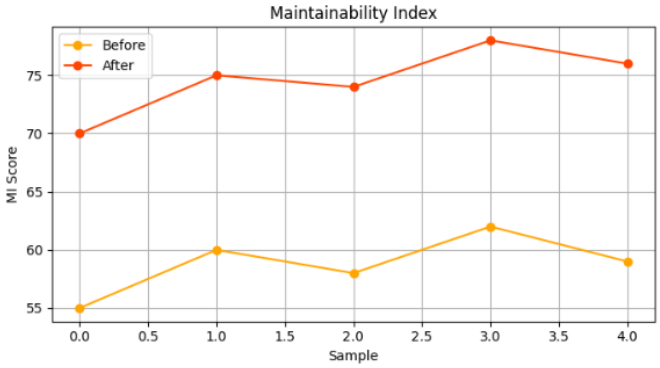


Fig. 3. MI Scores before and after TICRF.

Fig. 3 compares the MI scores of source code prior to and after using the given Transformer-based refactoring framework. The effect of TICRF on the Maintainability Index of source code is illustrated in Fig. 3. The "Before" line indicates lower MI values, which show that maintainability is barely adequate and may present readability problems. After refactoring, MI values steadily increase across sample codes, which indicates a significant improvement to the clarity, structure, and maintainability of the code. Therefore, TICRF improves the long-term maintainability of software.

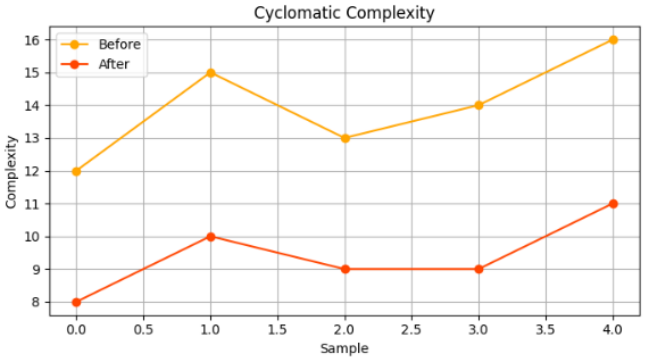


Fig. 4. Complexity scores before and after TICRF.



Fig. 5. Duplication rate before and after TICRF.

Fig. 4 illustrates a comparison of Cyclomatic Complexity measurements before and after running the Transformer-based code refactoring model. Fig. 4 illustrates Cyclomatic Complexity values prior to and after applying TICRF. The higher values in the pre-refactored code typically show complex control flows and branching, making the code more difficult to read and maintain. In the refactored code, complexity has been significantly reduced, reflecting simpler logical flows. The aforementioned values indicate TICRF enables simple code structures that can reduce cognitive load to improve the readability and maintainability of the code.

Fig. 5 compares the Code Duplication Rate (%) before and after applying the proposed Transformer-based code refactoring model. Fig. 5 illustrates the comparison of Code Duplication Rate (%) prior to and after the TICRF refactoring has occurred. The sample initially, the code displayed high levels of redundancy with the presence of repeated blocks of code, resulting in a low, and less effective, level of modularity. Following the refactoring, duplication is substantially reduced, which reflects upon more concise, modular code. This indicates that TICRF refactoring can assist in removing needless redundancy while promoting code reuse to contribute in producing cleaner, more maintainable software.

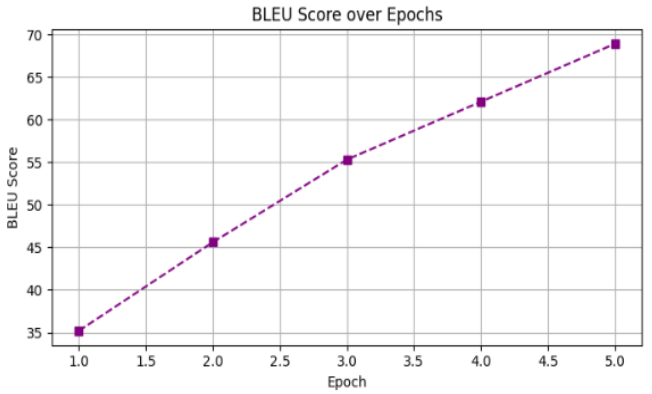


Fig. 6. BLEU Scores.

Fig. 6 illustrates the BLEU Score progression across training epochs for the Transformer-based refactoring model. Starting from an initial score of 35.2, the BLEU value steadily increases with each epoch, reaching 68.9 by the fifth epoch. This upward trend indicates that the model progressively learns to generate more accurate and syntactically aligned refactored code compared to the ground truth, validating the effectiveness of the fine-tuning process.

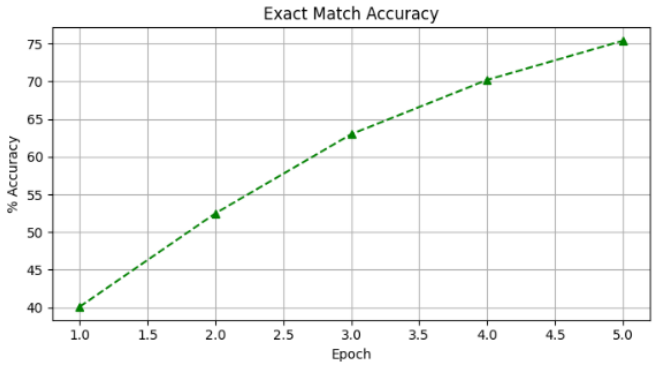


Fig. 7. Exact match accuracy.

Fig. 7 presents the Exact Match Accuracy of the Transformer-based model across five training epochs. The graph shows a steady increase from 40.1% in the first epoch to 75.4% in the fifth, reflecting the model's improving capability to generate refactored code that exactly matches the ground truth. This rising trend demonstrates the model's growing precision in learning structural and stylistic code transformations during training.

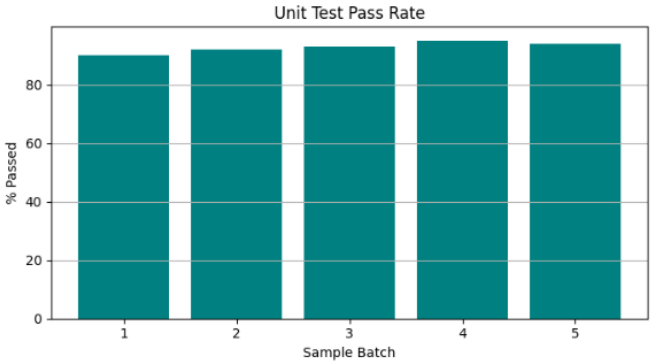


Fig. 8. Unit test pass rate.

Fig. 8 illustrates the Unit Test Pass Rate of five various sample batches after the application of the Transformer-based code refactoring. Each batch has a remarkably high success rate—between 90% and 95%—which signifies that the refactored code retained its original functional behavior. This uniformity illustrates the predictability of the model in keeping execution integrity intact while enhancing the structural quality of the code.

TABLE I. PERFORMANCE COMPARISON OF REFACTORING MODELS

Model	Precision	F1-Score	Accuracy	Recall
Decision Tree[20]	84%	85%	85%	86%
GNN[21]	91%	92%	92%	93%
Proposed Transformer Model	94%	97%	97%	98%

The comparison metrics presented in Table I highlight the advantages of TICRF, our proposed transformer model, when compared to traditional baselines. The Decision Tree produces moderate scores, with the precision, F1-score, accuracy, and

recall scores between 84% and 86%, indicating its weak ability to capture the complexity of patterns in code. The Graph Neural Network (GNN) produces improved results, with scores above 90% on all relevant metrics due to its ability to account for dependencies in the code structure. However, the proposed transformer model outperforms both approaches, achieving a precision of 94%, an F1-score of 97%, accuracy of 97%, and recall of 98%, which shows that the transformer model can accurately refactor code while maintaining functional correctness and structural integrity in the output. Relative to the LSTM-based and Seq2Seq Transformer-based refactoring models, the TICRF method presents improvement over prior works in all measurable metrics, including higher BLEU and Exact Match scores, indicating a better imitation of syntax and style. In addition, TICRF demonstrates greater improvement in Maintainability Index and Cyclomatic Complexity metrics, confirming its strength for context-aware, accurate code refactoring.

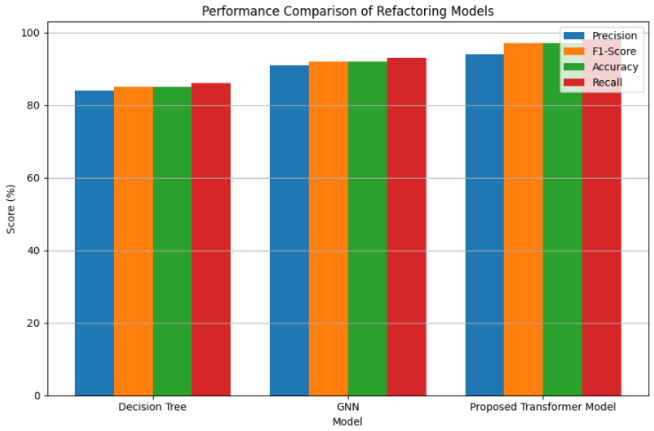


Fig. 9. Performance comparison.

Fig. 9 illustrates a performance comparison of three models of refactoring—Decision Tree, GNN, and Proposed Transformer Model—based on four measures: Precision, F1-Score, Accuracy, and Recall. The Decision Tree model performs moderately, with all measures in the mid-80% category. The GNN indicates significant improvement in all measures with a score of over 90%. The Proposed Transformer Model attains the best scores, with remarkable achievement of 94% Precision, 97% F1-Score and Accuracy, and 98% Recall. These outcomes confirm the excellence of the Transformer-based technique in accurately and effectively refactoring source code. To demonstrate real-world applicability, TICRF was used on a small open-source project written in Python (about 5,000 LOC). The model refactored functions that contained nested loops, duplicate code, and inconsistent naming conventions automatically. Developers who reviewed the refactored code indicated improved readability and maintainability without altering functionality, verifying the usefulness of the framework in practice rather than just on a curated dataset. The unit test pass rate of the project remained at 91% after the refactoring.

B. Discussion

The entire discussion of this research revolves around assessing the efficacy of the designed TICRF against conventional and contemporary baseline models. Under rigorous experimentation, it was seen that the Transformer-based model considerably improves several parameters related to the quality of code [22]. Traditional models such as the Decision Tree, while helpful for simple decision rules, do not have the contextualization necessary for advanced code transformation, as evidenced in their mid-scores on all measures [23]. The GNN model with graph-based structural information performs better through its capability of catching dependencies in code. Yet, it is still lacking compared to the Transformer model, which uses attention mechanisms to understand both semantic and syntactic patterns of source code.

The fine-tuned optimized CodeT5 model not only generates more maintainable code—as shown by increased Maintainability Index scores—but also decreases Cyclomatic Complexity and Code Duplication Rate significantly. The BLEU and Exact Match score improvements over epochs validate the model's capacity to learn efficient refactoring techniques. Moreover, the 90%+ Unit Test Pass Rate across all epochs ensures that the refactored code is functionally sound, an important requirement in software development. These improvements are also confirmed by quantitative measures, as the suggested model gained 94% precision, 97% accuracy and F1-score, and 98% recall. In conclusion, the Transformer-based solution is an effective and trustworthy tool for code refactoring automation, tackling both structural and semantic difficulties better than other models.

V. Conclusion and Future Works

A robust and intelligent approach to automated code refactoring using a Transformer-based model, specifically leveraging the capabilities of the fine-tuned CodeT5 model, has been introduced. Traditional approaches, such as decision trees and graph neural networks, while effective to some extent, fail to capture the deeper syntactic and semantic complexities of source code. The suggested framework addresses such limitations via a sequence-to-sequence architecture that transforms poorly organized code into better-maintainable and optimized forms. From dataset preparation and preprocessing to model training and inference, the pipeline ensures both structural and functional correctness of refactored output via AST validation and extensive testing. Experimental outcomes show conspicuous enhancements in significant software quality metrics like Maintainability Index, Cyclomatic Complexity, and Code Duplication Rate. The Transformer model has a precision of 94%, recall of 98%, and an accuracy of 97%, comparing favorably with baseline approaches. Overall, this work showcases the transformative value of Transformer-based models for enhancing code quality, mitigating technical debt, and allowing programmers to keep systems cleaner, scalable, and more reliable.

As future directions, the Transformer-based refactoring framework can be extended to include multilingual code support to process more types of programming languages. The inclusion of semantic-aware embeddings and graph-based structural constraints in addition to Transformer outputs would help the model to be even more accurate in identifying and rewriting intricate code patterns. Real-world deployment in development environments like IDEs with adaptive feedback loops would increase interactivity and context awareness. In addition,

reinforcement learning methods may be investigated to create dynamic, optimal refactoring approaches to optimize long-term maintainability and performance results.

REFERENCES

[1] V. Veeramachaneni, "AI-driven software development: enhancing code quality and maintainability through automated refactoring," Int J Recent Dev Sci Technol, vol. 5, no. 6, pp. 94–101, 2021.

[2] A. Ainapure, A. Kharote, T. Agrawal, and S. Dhage, "Automate Code Refactoring for Enhanced Software Maintenance and Development".

[3] M. R. Lyu, B. Ray, A. Roychoudhury, S. H. Tan, and P. Thongtanunam, "Automatic programming: Large language models and beyond," ACM Trans. Softw. Eng. Methodol., vol. 34, no. 5, pp. 1–33, 2025.

[4] K. Jain, "Exploiting Test Structure to Enhance Language Models for Software Testing," PhD Thesis, Carnegie Mellon University, 2025.

[5] J. Cordeiro, S. Noei, and Y. Zou, "An Empirical Study on the Code Refactoring Capability of Large Language Models," ArXiv Prepr. ArXiv241102320, 2024.

[6] L. LEMNER and L. WAHLGREN, "Predicting the Need for Test Maintenance Using LLM Agents-Applying Test Maintenance Factors to Changes in Production Code to Identify If and Where Test Cases Need to Be Updated," 2024.

[7] M. Dewey, "Large Language Models and Software Testing," 2024.

[8] A. Rehman, "AI Driven Code Review System: Leveraging Artificial Intelligence for Enhanced code Quality Assessment and Bug Detection," 2025.

[9] K. Tsybulka, "Enhancing code quality through automated refactoring techniques," PhD Thesis, ETSI_Informatica, 2024.

[10] L. Lemner, L. Wahlgren, G. Gay, N. Mohammadiha, J. Liu, and J. Wennerberg, "Exploring the Integration of Large Language Models in Industrial Test Maintenance Processes," ArXiv Prepr. ArXiv240906416, 2024.

[11] A. Aljohani and H. Do, "From fine-tuning to output: An empirical investigation of test smells in transformer-based test code generation," in Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing, 2024, pp. 1282–1291.

[12] M. A. H. M. A. Hodovychenko and D. D. K. D. D. Kurinko, "Analysis of existing approaches to automated refactoring of object-oriented software systems," Вісник Сучасних Інформаційних Технологій, vol. 8, no. 2, pp. 179–196, 2025.

[13] K. Parvathinathan et al., "Automated Unit Testing Frameworks for Deep Learning Components in ML Software Stacks," 2025.

[14] G. Bandarupalli, "AI-driven code refactoring: Using graph neural networks to enhance software maintainability," ArXiv Prepr. ArXiv250410412, 2025.

[15] A. S. Yaraghi, D. Holden, N. Kahani, and L. Briand, "Automated test case repair using language models," IEEE Trans. Softw. Eng., 2025.

[16] M. Metsola, "LARGE LANGUAGE MODELS ON SOFTWARE REFACTORING," 2024.

[17] A. Barzilay, "Using natural language processing techniques for automated code refactoring," PhD Thesis, Universidade de SÃ\pounds o Paulo, 2023.

[18] I. Palit and T. Sharma, "Generating refactored code accurately using reinforcement learning," ArXiv Prepr. ArXiv241218035, 2024.

[19] O. Duggineni, "CodeSearchNet," 2023, [Online]. Available: https://www.kaggle.com/datasets/omduggineni/codesearchnet

[20] Y.-Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," Shanghai Arch. Psychiatry, 2015.

[21] V. Gadey, R. Goetz, C. Sendner, S. Sovio, and A. Dmitrienko, "GNN-Based Code Annotation Logic for Establishing Security Boundaries in C Code," ArXiv Prepr. ArXiv241111567, 2024.

[22] E. A. Issawi and O. A. Hajjouz, "Comparative Analysis of Conventional Approaches and AI-Powered Tools for Unit Testing within Web Application Development," LU-CS-EX, 2024.

[23] E. Sundqvist, "AI-Assisted Unit Testing: Empirical Insights into GitHub Copilot Chat's Effectiveness and Collaborative Benefits." 2024.

Real-Time Biomechanical Squat and Deadlift Posture Analysis Using Google Machine Learning Kit

AI-Powered Fitness Coach

Liew Yee Jie[1], Ting Tin Tin[2]*, Chaw Jun Kit[3], Ali Aitizaz[4]*, Ayodeji Olalekan Salau[5a, b], Omolayo M. Ikumapayi[6], Lim Siew Mooi[7]

Faculty of Computing and Information Technology, Tunku Abdul Rahman University of Management and Technology, Kuala Lumpur, Malaysia[1, 7]

Faculty of Data Science and Information Technology, INTI International University, Nilai, Malaysia[2, 5, 6]

Institute of Visual Informatics, Universiti Kebangsaan Malaysia, Bangi, Malaysia[3]

School of Technology, Asia Pacific University, Malaysia[4]

School of IT, UNITAR International University, Malaysia[2, 4]

Department of Electrical/Electronics and Computer Engineering, Afe Babalola University, Ado, Ekiti, Nigeria[5a]

Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu, India[5b]

Department of Mechanical and Industrial Engineering, University of Johannesburg, Johannesburg, South Africa[6]

Department of Mechanical and Mechatronics Engineering, Afe Babalola University, Ado Ekiti, Nigeria[6]

***Abstract*—This project presents the development of a mobile application for real-time posture analysis during squat and deadlift exercises, using Google Machine Learning (ML) Kit pose detection. Proper exercise form is critical in preventing injuries, underscoring the need for systems that provide immediate feedback, an aspect often missing in existing fitness applications. This study addresses that gap by designing an app that not only guides users through motion analysis but also incorporates a safety mechanism to detect sudden falls. The system employs algorithms to process landmarks, calculate joint angles, count repetitions, and trigger emergency alerts. Two groups of bodybuilders confirmed the usability and accuracy in real-time biomechanical squat and deadlift posture analysis. These findings contribute to the field of AI-driven fitness by introducing a non-wearable, mobile-based solution for guided strength training. In addition, it offers societal benefits as an AI-powered fitness coach that aims to promote public health.**

***Keywords*—Pose detection; squat and deadlift; Google ML Kit; fitness; posture analysis; emergency; public health**

I. INTRODUCTION

The evolution of technology within the fitness industry has significantly advanced training methods and monitoring capabilities for athletes, particularly in strength training disciplines such as bodybuilding and powerlifting. However, experts such as coach / trainer are required to provide real-time feedback during the training to ensure the accuracy and efficiency of training [1,2]. This is an issue for those who cannot afford a high-paying coach or lack of sufficient coach for everyone. Furthermore, common injuries in strength training, which often result from improper form, continue to be a major concern [3,4]. These injuries not only hinder progress but can also lead to long-term health issues, significantly impacting an athlete's training and competitive performance.

Current fitness applications predominantly emphasise quantitative metrics such as calories burnt, distance covered, or heart rate variability. While these data points are valuable, they fail to capture the qualitative dimensions of strength training exercises, such as joint angles during a squat or spinal alignment during a deadlift. These qualitative aspects are critical to ensure proper form and minimizing the risk of injury. Consequently, there is a compelling need for more specialised technology that provides real-time, context-aware feedback on exercise form and technique.

This project presents a novel mobile application that utilises real-time motion capture technology to improve exercise performance and safety in strength training. By integrating Google's ML Kit pose detection for accurate skeletal tracking, the application delivers immediate, actionable feedback to users during their workouts. This capability not only supports injury prevention but also improves training efficiency by ensuring correct biomechanics are maintained throughout each exercise. The innovative design and user-centric interface address a significant weakness in existing fitness solutions, which often neglect real-time biomechanical evaluation. By bridging this gap, the proposed system empowers users to train more effectively, safely and autonomously.

This paper is constructed in five sections. Section II summarises the existing research studies and market products of fitness assistance and highlights the research gap. Section III presents the proposed architecture of the system with associating algorithms. Section IV records the performance of the proposed AI-powered fitness coach using an experiment carried out among bodybuilders. Finally, Sections V and VI discuss and compare this research with previous research and market products, highlight the limitations of the research and future works.

*Corresponding authors

This article processing costs are supported by INTI International University and Asia Pacific University, Malaysia.

II. Literature Review

This review of the literature systematically examines key areas relevant to the development of a real-time posture analysis application tailored for squat and deadlift exercises. It begins with an overview of motion capture technologies, including both optical and non-optical systems, and traces their evolution with the integration of artificial intelligence - highlighting the enhanced analytical capabilities brought about by these advancements. The review then focusses on Google's ML Kit for pose detection, highlighting its role in enabling real-time feedback mechanisms that are critical for immediate posture correction in exercise applications.

A detailed exploration of existing applications that employ Google's ML Kit illustrates its practical implementation and growing adoption in domains such as fitness, yoga, and posture correction. Furthermore, the review delves into the biomechanics of squats and deadlifts, discussing correct form, joint alignment, and load management, thus emphasising the importance of precise execution to prevent injury and optimise performance.

The analysis also evaluates current fitness applications, identifying their core functionalities and limitations compared to the proposed system. In particular, the proposed application differentiates itself by offering advanced analytics, AI-driven real-time posture feedback, and built-in emergency response mechanisms. Each component of the review is structured according to the PRISMA methodology, ensuring a comprehensive, systematic, and evidence-based foundation that supports the rationale and design of the proposed solution.

A. Background of Motion Capture Technology

Motion capture (MoCap) refers to the digital tracking and recording of movement patterns of objects or living beings within a defined space, using a variety of technologies and methodologies [5]. It is widely used in domains such as the military, entertainment, sports, medicine, and the validation of computer vision and robotic systems [6]. Broadly, motion capture systems are classified into two categories: optical and non-optical.

Optical systems typically use cameras in conjunction with reflective markers or light-emitting diodes (LEDs) to capture motion as shown in Fig. 1 [7]. These systems are renowned for their high precision and are widely used in industries such as film production, video game development, and sports performance analytics [8,9]. On the contrary, non-optical systems—such as inertial measurement units (IMUs)—employ embedded sensors to track motion independently of camera setups, offering enhanced mobility and adaptability across diverse environments [10]. Fig. 2 is an example of non-optical system [11]. Despite their differing methodologies, both types can deliver high levels of accuracy and precision in motion tracking.

Recent advances have seen the integration of artificial intelligence (AI) with optical motion capture technologies, significantly enhancing their analytical and functional capabilities. AI-driven models enable more advanced data interpretation and facilitate real-time feedback mechanisms, which are crucial for dynamic applications [12]. This fusion of AI and motion capture not only improved the efficiency and reliability of motion analysis but has also broadened its applicability to fields such as medical diagnostics, personalised fitness training, and rehabilitation, where accurate real-time movement assessment is essential.



Fig. 1. Optical system [7].

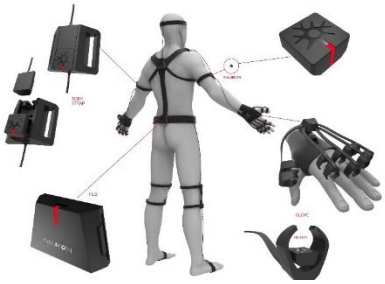


Fig. 2. Non-optical system (Inertial system) [11].

B. Recent AI Motion Capture Technology

Recent advances in AI-powered motion capture technologies have significantly improved the precision, accessibility, and adaptability of these systems in various environments. Current innovations primarily aim to improve the speed and accuracy of pose estimation through deep learning models, while expanding the application beyond its traditional domains. One notable milestone in this evolution is the introduction of Deep3DPose, a pioneering model that applies deep learning to human pose estimation. Using cascaded deep neural networks, Deep3DPose directly infers human poses from images and introduces a multitask framework capable of concurrently predicting multiple outputs, thereby enabling accurate 3D pose estimation and full-body reconstruction in real time [13]. Fig. 3 shows an example of AI motion capture technology introduced in the Brau et al. (2016) research.

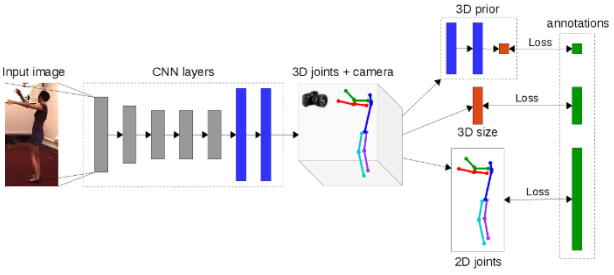


Fig. 3. AI motion capture technology captures human pose and convert it into 3D joints [14].

Subsequent research has introduced numerous algorithms aimed at addressing the limitations of conventional motion capture systems. These include enhanced marker-based approaches [15], hybrid architectures such as Convolutional

Neural Networks (CNNs) combined with Long Short-Term Memory (LSTM) models [16], and frameworks grounded in Human-Centred Artificial Intelligence (HAI) that prioritize interpretability and user-specific customisation [17]. Collectively, these developments signify a paradigm shift towards more intelligent, robust, and user-centric motion capture technologies, with implications for fields such as rehabilitation, sports science, ergonomics, and interactive systems.

C. Google's ML Kit for Pose Detection

Within the domain of AI-driven motion capture technologies, Google's ML Kit for pose detection emerges as a crucial tool, enabling the efficient development of mobile applications capable of real-time human pose estimation. The ML Kit provides a flexible and accessible framework for embedding machine learning functionalities directly into mobile applications on both the Android and iOS platforms. It includes a comprehensive set of APIs that support the detection and tracking of human body joints from static images and live video streams [18].

The pose detection API in the ML Kit is engineered to recognize 33 anatomical landmarks in the human body, including major joints such as shoulders, elbows, wrists, hips, knees, and ankles, as illustrated in Fig. 4 [19]. This level of granularity enables the development of fitness and wellness applications capable of assessing user posture and movement patterns. In particular, it supports real-time form analysis and corrective feedback for strength training exercises such as squats and deadlifts, delivering functionality comparable to that of more complex and computationally intensive motion capture systems, but with significantly lower resource demands and greater accessibility.

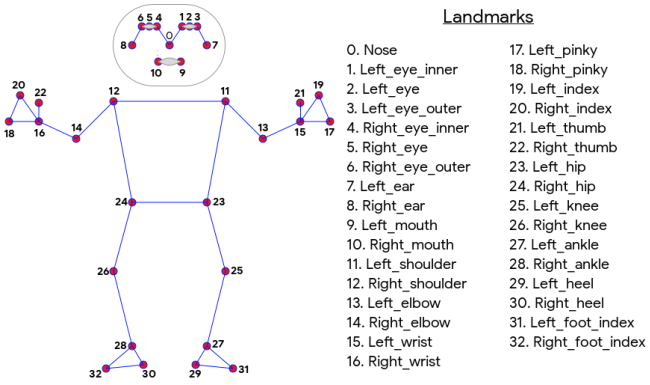


Fig. 4. 33 landmarks of a human pose [19].

A distinguishing feature of Google's ML Kit is its dual-mode operational capability. It supports both a high-accuracy mode, which utilises cloud-based processing and requires an active internet connection, and a real-time mode that functions entirely offline to ensure minimal latency. This flexibility is particularly advantageous for applications in real-time sports analytics, fitness coaching, and augmented reality, where immediate and continuous feedback based on user movement is critical. By accommodating both high-performance and resource-constrained environments, ML Kit broadens its applicability across a wide range of mobile use cases. Fig. 5 shows an example of application using Google's ML Kit landmarks no. 11-18. Sample angles calculations are shown Fig. 6.

Landmark	Type	Position	InFrameLikelihood
11	LEFT_SHOULDER	(734.9671, 550.7924, -118.11934)	0.9999038
12	RIGHT_SHOULDER	(391.27032, 583.2485, -321.15836)	0.9999894
13	LEFT_ELBOW	(903.83704, 754.676, -219.67009)	0.9836427
14	RIGHT_ELBOW	(322.18152, 842.5973, -179.28519)	0.99970156
15	LEFT_WRIST	(1073.8956, 654.9725, -820.93463)	0.9737737
16	RIGHT_WRIST	(218.27956, 1015.70435, -683.6567)	0.995568
17	LEFT_PINKY	(1146.1635, 609.6432, -956.9976)	0.95273364
18	RIGHT_PINKY	(176.17755, 1065.838, -776.5006)	0.9785348

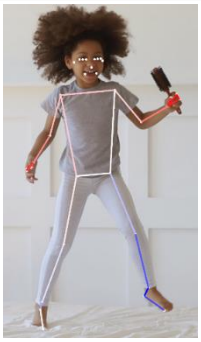


Fig. 5. Example of Google's ML Kit for pose estimation [19].

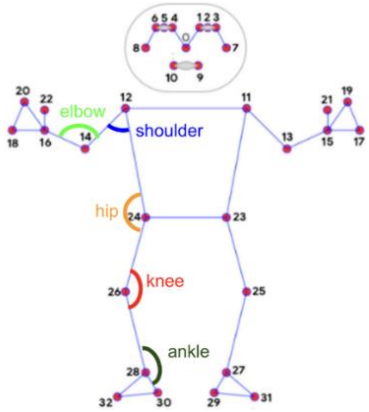


Fig. 6. Calculate the angle between landmarks [19].

D. Applications Using Google's ML Kit for Pose Detection

Google's ML Kit Pose Detection API has seen widespread adoption in mobile application development in both academic research and industry practice. Numerous projects, from student-level prototypes to fully deployed applications, have demonstrated their effectiveness in tracking human posture and interpreting movement in domains such as fitness, yoga, and dance. These implementations underscore the practicality of the ML Kit as a lightweight, on-device solution for real-time pose analysis, even on resource-constrained devices.

A notable academic implementation is the undergraduate thesis by Nguyen (2024), which investigated the integration of ML Kit Pose Detection into Android applications [20]. The project resulted in two mobile applications: one designed for detecting static body poses from images and another for real-

time yoga pose recognition using a custom-trained classifier. Using the detection of the landmarks of the pose of the ML Kit, the system successfully identified common yoga postures - including tree pose, downward dog, and plank - demonstrating the API's potential of the API to support mobile health and wellness applications through computer vision–driven feedback.

Another example is RepDetect, a real-time fitness assistant developed by students at Metropolia University of Applied Sciences. This Android-based application uses the ML Kit to monitor exercises such as push-ups, sit-ups, and lunges by evaluating the consistency of the pose and counting repetitions [35]. The app uses pose confidence scores to assess form accuracy, providing users with real-time feedback and exercise statistics. This project exemplifies how small development teams can leverage the ML Kit and Kotlin to create fully functional, AI-driven workout tracking solutions.

In industry settings, the ML Kit has also been incorporated into commercial applications. Groovetime, a dance-based mobile app, utilises pose estimation to help users learn choreographed routines by comparing user movements against predefined dance sequences. It offers real-time feedback on movement accuracy, acting as a virtual dance coach [21]. Similarly, NeckFit uses the ML Kit to analyse upper-body posture and suggest corrective exercises aimed at alleviating neck strain and improving spinal alignment. Designed for daily use, NeckFit targets users seeking ergonomic improvements and posture correction [22].

Together, these examples reflect the growing utility and adaptability of ML Kit Pose Detection in diverse application areas, reaffirming its value as a robust, accessible tool for real-time, mobile-based human pose analysis.

E. Squat

Squat is a fundamental compound exercise that engages multiple muscle groups in the body, including the quadriceps, hamstrings, glutes, lower back, and core. The movement involves lowering the body into a squatting position by flexing the knees and hips while maintaining a neutral spine, followed by returning to a standing position. Known for its effectiveness in enhancing lower body strength, flexibility, and balance, the squat is, however, frequently performed incorrectly in gym settings. Improper execution can result in a variety of injuries, such as lower back pain, muscle strains or tears, knee sprains, and tendinopathy, an overuse injury commonly associated with repetitive movements [1].

Fig. 7 demonstrates proper squat execution: place the barbell on the trapezius muscle and hold it in a comfortable hand position. The feet should be slightly wider than shoulder-width apart with toes pointing outward, ideally not exceeding 10 degrees [23]. As you descend into the squat, ensure that your trunk remains parallel to your tibia to maintain balance and alignment. The depth of the squat and the angle of knee flexion should correspond to the specific type of squat being performed: mini squats typically range between 140° and 150°, semi-squats between 120° and 140°, half-squats between 80° and 110°, and deep squats go below 80° [24]. Precise joint angles are crucial to prevent injuries and optimise the exercise's effectiveness: hip joints should ideally be angled at approximately 58.0°±9.8°, ankle joints at about 81.0°±7.3°, and the torso at roughly 38.2°±5.8° [25, 26]. These biomechanical considerations help to perform squats correctly and safely, reducing the risk of injuries while improving training outcomes.

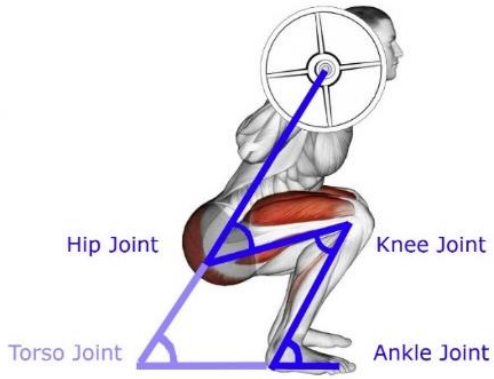


Fig. 7. Proper squat pose.

According to Strength Level (2024), the selection of appropriate dumbbell weights for squats is influenced by three primary factors: strength level, body weight, and age [27]. As illustrated in Fig. 8, the advised weight varies significantly between different strength levels. For example, a beginner who weighs 60 kg is typically advised to use lighter weights, which are 45 kg (0.75x body weight), to promote correct form and ensure safety. Additional guidance is provided in Fig. 9 and Fig. 10, which delineate the recommended weights according to body weight and age, respectively, helping to tailor the exercise to the individual's physical condition and reducing the risk of injury.

Strength Level	Bodyweight Ratio
Beginner	0.75x
Novice	1.25x
Intermediate	1.50x
Advanced	2.25x
Elite	2.75x

Fig. 8. Advice weight for dumbbell for different strength level for squats [27].

Bodyweight	Beginner	Novice	Intermediate	Advanced	Elite
50	33	52	76	104	136
55	40	60	86	116	149
60	47	68	95	127	161
65	53	76	104	137	173
70	59	83	113	147	184
75	66	91	122	157	195
80	72	98	130	166	205
85	78	105	138	175	215
90	83	112	146	184	225
95	89	118	153	192	234
100	95	125	160	201	243
105	100	131	168	209	252
110	106	137	174	216	260
115	111	143	181	224	269
120	116	149	188	231	277
125	121	155	194	238	284
130	126	160	201	245	292
135	131	166	207	252	299
140	136	171	213	259	307

Fig. 9. Advice weight (kg) for dumbbell according to different body weight (kg) and strength level for squats [27].

Age	Beginner	Novice	Intermediate	Advanced	Elite
15	55	80	111	147	187
20	62	91	127	168	214
25	64	93	130	173	219
30	64	93	130	173	219
35	64	93	130	173	219
40	64	93	130	173	219
45	61	89	123	164	208
50	57	83	116	154	195
55	53	77	107	142	180
60	48	70	98	130	165
65	44	63	88	117	149
70	39	57	79	105	134
75	35	51	71	94	119
80	31	46	63	84	107
85	28	41	57	75	96
90	25	37	51	68	86

Fig. 10. Advice weight (kg) for dumbbell according to different age and strength level for squats [27].

F. Deadlift

The deadlift is a fundamental weight training exercise that engages multiple muscle groups, making it one of the most effective compound movements to build overall strength. Primarily, deadlift targets the glutes, hamstrings, lower back, and core, while also activating the forearms and traps [28]. The exercise involves lifting a loaded barbell from the ground to the level of the hips, followed by a controlled lowering of the weight back to the floor. Recognised for its ability to enhance muscular strength and hypertrophy, the deadlift is also valued for improving grip strength and overall body stability.

Fig. 11 shows several variations of the deadlift, including the conventional deadlift, sumo deadlift, straight-leg deadlift, and Romanian deadlift (RDL) [29]. However, improper technique during deadlifting can result in excessive compressive and shear forces on the lumbosacral spine, potentially leading to disc herniation or other spinal injuries [30].

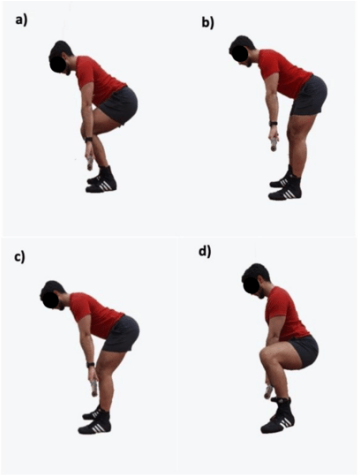


Fig. 11. a) Conventional deadlift, b) Straight-leg deadlift, c) Romanian deadlift, d) Sumo deadlift [29].

Proper execution of deadlifts varies according to the specific style being performed, with each variation emphasising distinct stances and grip positions. The conventional deadlift, as outlined by Escamilla et al. (2002), involves a stance narrower than shoulder width, with the feet positioned close to the centre of the bar and the hands gripping the bar just outside the legs [31]. Straight-leg deadlift is similar to the Romanian deadlift, but differs by maintaining fully extended knees and ensuring that the bar does not come into contact with the legs. On the contrary, the Romanian deadlift retains the stance width of the conventional deadlift but limits knee flexion to no more than 15 degrees, ensuring that the barbell stays in contact with the legs throughout the movement [32]. Lastly, the sumo deadlift features a significantly wider stance, exceeding shoulder width, which facilitates the participation of different muscle groups [31].

In terms of determining the appropriate weight for the deadlift, the guidelines provided by Strength Level Limited (2024) emphasise the importance of adjusting weight based on an individual's strength level, body weight, and age. As illustrated in Fig. 12, there is considerable variation in recommended deadlift weights in different strength categories. For example, a beginner weighing 60 kg is typically advised to lift a weight equal to your body weight (60 kg) to ensure proper form and reduce the risk of injury. Additional recommendations are provided in Fig. 13 and 14, offering customised weight guidelines based on body weight and age, respectively. These individualised suggestions enable users to adjust their training to their physical abilities, thereby promoting safety and reducing the likelihood of injury.

Strength Level	Bodyweight Ratio
Beginner	1.00x
Novice	1.50x
Intermediate	2.00x
Advanced	2.50x
Elite	3.00x

Fig. 12. Advice weight for dumbbell for different strength level for deadlift [27].

Bodyweight	Beginner	Novice	Intermediate	Advanced	Elite
50	44	65	93	125	160
55	51	74	103	137	174
60	58	83	114	149	187
65	66	92	124	160	200
70	73	100	133	171	212
75	79	108	142	182	224
80	86	116	151	192	235
85	93	123	160	201	245
90	99	131	168	211	256
95	105	138	176	220	266
100	111	145	184	228	275
105	117	151	192	237	284
110	123	158	199	245	293
115	129	164	206	253	302
120	134	171	213	261	311
125	140	177	220	268	319
130	145	183	227	276	327
135	150	188	233	283	335
140	155	194	240	290	342

Fig. 13. Advice weight (kg) for dumbbell according to different body weight (kg) and strength level for deadlift [27].

Age	Beginner	Novice	Intermediate	Advanced	Elite
15	67	95	130	170	213
20	76	109	148	194	244
25	78	112	152	200	250
30	78	112	152	200	250
35	78	112	152	200	250
40	78	112	152	200	250
45	74	106	145	189	238
50	70	99	136	178	223
55	65	92	125	164	206
60	59	84	115	150	188
65	53	76	103	135	170
70	48	68	93	122	153
75	43	61	83	109	136
80	38	54	74	97	122
85	34	49	67	87	109
90	31	44	60	79	99

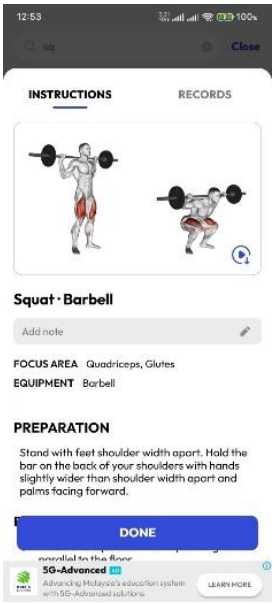
Fig. 14. Advice weight (kg) for dumbbell according to different age and strength level for deadlift [27].

G. Existing Product

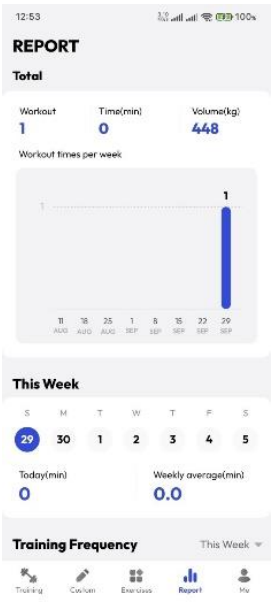
1) Gym workout tracker: Gym Log: This fitness app, developed by Leap Fitness Group, has amassed more than 5 million downloads and boasts a perfect 5.0/5.0 rating from 92.8k reviewers on the Google Play Store [33]. Launched on December 8, 2021, the app is designed to serve beginners and experienced bodybuilders by offering customisable workout routines that align with individual goals and available gym equipment. It functions as a comprehensive fitness tool, providing access to a robust database of more than 500 exercises, each accompanied by detailed visual and written instructions (Fig. 15). In addition to facilitating workout planning and tracking, the app replaces traditional pen-and-paper methods by allowing users to log workouts digitally. It also tracks progress through intuitive statistics and charts, offering users a detailed overview of their performance. Features such as adjustable rest timers, unlimited routine edits, and the ability to create personalised workout routines ensure a highly flexible and enriched training experience. By combining these functions, the app helps users optimize their workouts, improve training efficiency, and maintain motivation, all without the expense of a personal trainer.



(a) Gym log app



(b) Guidance page



(c) Report page

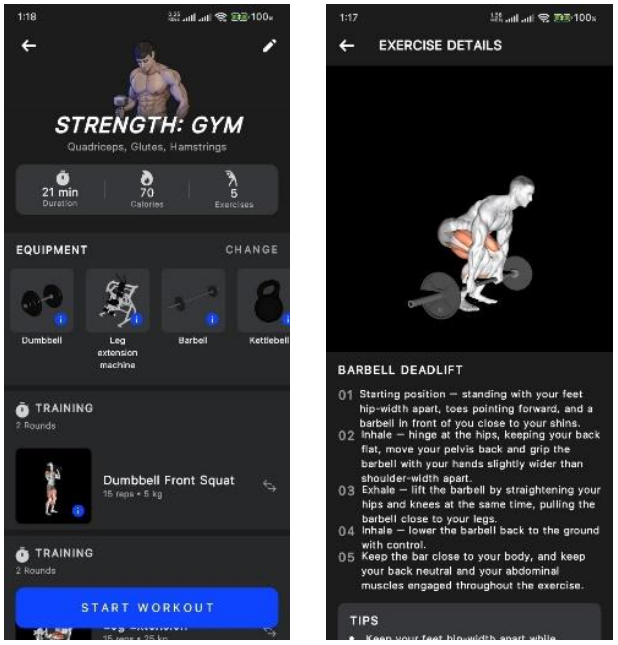
Fig. 15. Gym Log mobile apps [33].

2) Muscle booster – Plan workouts: With more than 10 million downloads and a rating of 4.8/5.0 from users on the Google Play Store, Muscle Booster is a highly rated fitness app developed by Welltech Apps Limited on November 27, 2019 [34]. This dynamic and comprehensive fitness assistant is designed to cater to both men and women, offering customisable workout plans tailored to individual fitness goals, equipment availability, and personal constraints.

Muscle Booster is versatile enough to be used both at home and in the gym, featuring a vast library of over 1,000 workouts. It also includes interactive challenges and provides detailed guidance through a workout player that offers audio tips and timed instructions. Whether users are looking to gain muscle, lose weight, or focus on recovery, the app allows them to set specific objectives, target muscle zones, and receive personalised recommendations. It also incorporates mini milestones to track progress, helping users stay motivated throughout their fitness journey. As a result, Muscle Booster serves as an all-encompassing platform to improve physical fitness and facilitating health transformations efficiently (Fig. 16).



(a) Icon of the muscle booster – Plan workouts



(b) Gym planner (c) Guidance page

Fig. 16. Muscle booster mobile apps [34].

H. Summarise of Existing Product

In summary, while apps like Gym Workout Tracker: Gym Log and Muscle Booster – Plan Workouts primarily focus on providing extensive exercise libraries and detailed instructional content, this project distinguishes itself by incorporating advanced real-time posture analysis. This feature is particularly valuable as it not only guides users through exercises but also offers instant feedback on their form, thereby improving both safety and effectiveness.

TABLE I. COMPARISON OF EXISTING PRODUCT WITH THIS PROJECT

Criteria	*Gym Workout Tracker*	*Muscle Booster*	*This project*
Registration	✓	✓	✓
Login	✓	✓	✓
Profile	✓	✓	✓
Custom workout	✓	✓	✓
Workout planner	✓	✓	✓
Weight tracker	✓	✓	✓
Workout weight recommender			✓
AI Real-time posture analysis with feedback			✓
Advanced analytics			✓
Emergency mechanism			✓

This project further differentiates itself through the integration of more sophisticated technology, specifically Google's ML Kit for pose detection, something existing apps do not utilize to the same extent. This technology enhances the accuracy of posture and form analysis, representing a significant advancement in the app's capabilities. Additionally, the emergency mechanism is a unique safety feature designed to detect potential distress or unsafe conditions during workouts, triggering a prompt response. This feature is especially noteworthy in the fitness app market, where similar safety features are typically reserved for expensive high-tech fitness devices, such as smartwatches. Table I summarises the comparison of this project with existing products found in the market focusing on features comparison.

III. MATERIAL AND METHOD

A. System Architecture Design

This section presents the overall system architecture design that includes 1) real time input, 2) pose analysis, and 3) real time pose adjustment. For real time input, the Google machine learning kit is used to detect and extract key joint landmarks from the user's body. These coordinates are used in later calculations for the classification of angles, positions, and postures. Pose analysis is carried out using angles between joints calculation and detects incorrect reps (posture validation). If there are emergencies, such as injuries or falls, an AI-powered fitness coach will alert surrounding users to help in the emergency. In the event of incorrect posture, the system will alert the user, too. A workout plan is incorporated into this system to monitor the duration of each workout. The details of each component will be further explained using workflow, and algorithm will be illustrated using pseudocode in the following subsections (Fig. 17).

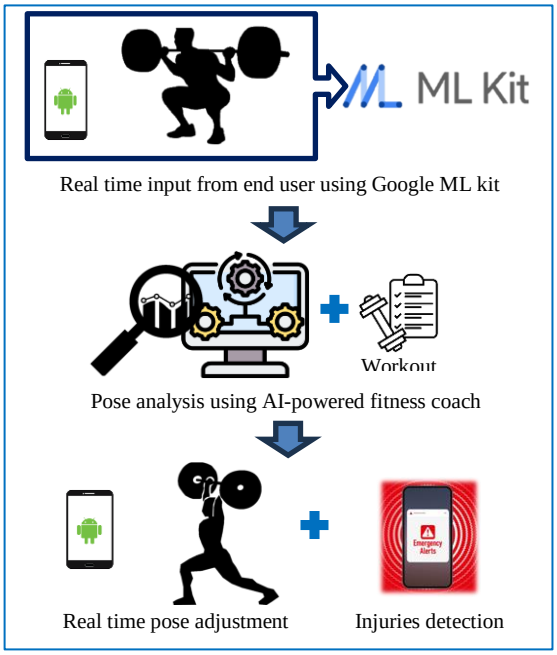


Fig. 17. AI-powered fitness coach system architecture design.

B. System Components Workflow Design

This section presents three flowcharts designed to support real-time biomechanical posture analysis for squats and deadlifts. The first flow chart (Fig. 18) outlines the process of capturing a user's workout plan based on their profile, followed by the generation of personalised exercise recommendations. The process begins when the user navigates to the workout planner page. The system retrieves and displays all existing workout

plans from the database. At this point, the user can choose to create a new plan, modify an existing one, or delete a plan. When creating a new workout plan, the system displays a recommended weight based on the user profile. The user then inputs key details of the working of the exercises, including selected sets, weights, and repetitions. Before finalising the plan, a confirmation prompt ensures that the user reviews the input. Once confirmed, the workout plan is stored in the database, and a success dialogue notifies the user of the successful creation.

The modification process mirrors the creation workflow. After selecting a plan to modify, the system again displays recommended weights. Users can update the exercise details and confirm the changes. Upon validation, the updated plan is saved in the database and a confirmation dialog is displayed. For deletion, the user is asked to confirm the action to prevent accidental loss. If confirmed, the selected workout plan is removed from the database and a success dialog provides final confirmation.

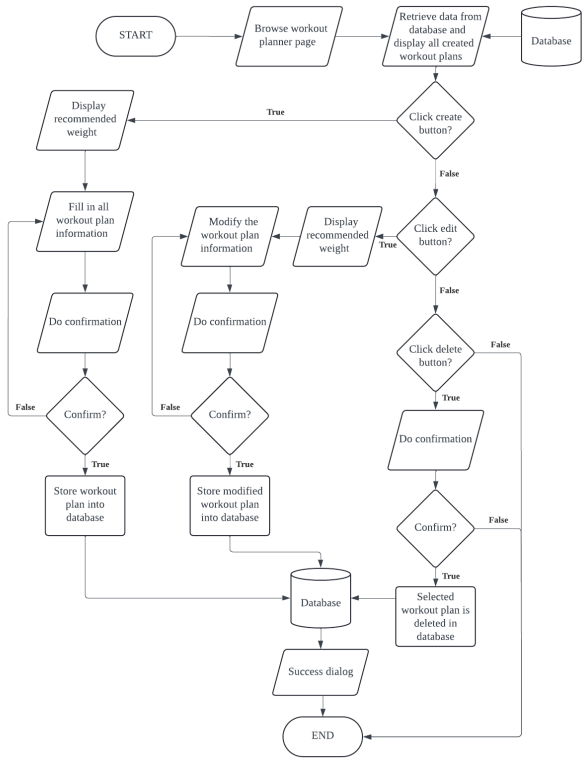


Fig. 18. Flow chart for the exercise planner module.

Fig. 19 illustrates the flow chart for the motion capture module. The process begins when a user accesses the workout planner page and selects a workout plan. The application then displays relevant instructions and tips based on the selected workout, including information on necessary equipment and targeted muscle groups. If the user chooses to calculate the appropriate barbell weight, they can utilise the built-in plate calculator by entering the desired total weight and selecting the available weight plates. The application then generates the optimal plate configuration for the workout. Once the user starts the workout, the application becomes a real-time monitoring interface that captures and analyses the user's movements. Using motion capture technology, the system provides immediate feedback on exercise form, helping the user make real-time adjustments to maintain proper posture and technique.

During the workout session, users can add additional sets if needed. The system also includes an embedded emergency detection mechanism that continuously monitors for abnormal activity, such as sudden falls. If an emergency is detected, the application triggers an audible alarm, notifies the designated emergency contact, and offers the user the option to stop the alert. Upon completion of all sets or if the user chooses to end the workout, the system stores the entire session's data, including performance metrics and form analysis, into the database. Finally, a congratulatory screen is displayed to acknowledge the effort before redirecting them back to the workout planner page for future session planning or review.

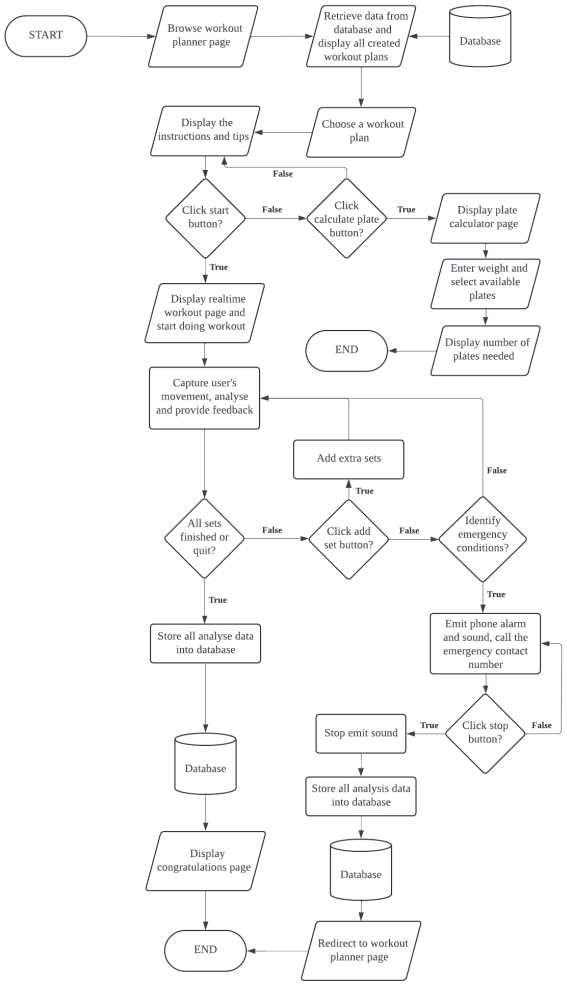


Fig. 19. Flowchart for motion capture module.

Fig. 20 delineates the process by which users interact with their exercise and body weight data within the fitness application. The process begins when a user accesses the report page, prompting the app to retrieve and display several types of data: an overall summary of the frequency and volume, detailed logs of individual workout sessions, and trends in the

progression of body weight. Users are presented with a weekly overview of all recorded workouts. For a more granular view, they can select the "View All" option, which opens a calendar interface that allows users to filter historical data by specific dates. Upon selecting a date, the application displays only the corresponding workout sessions, allowing users to examine detailed performance metrics, including repetitions, joint angles, and explosive power.

Alternatively, if the user chooses to log their current body weight rather than review historical data, the system prompts them to input the new weight value. These data are then securely stored in the database, contributing to long-term tracking and analysis of physical progress.

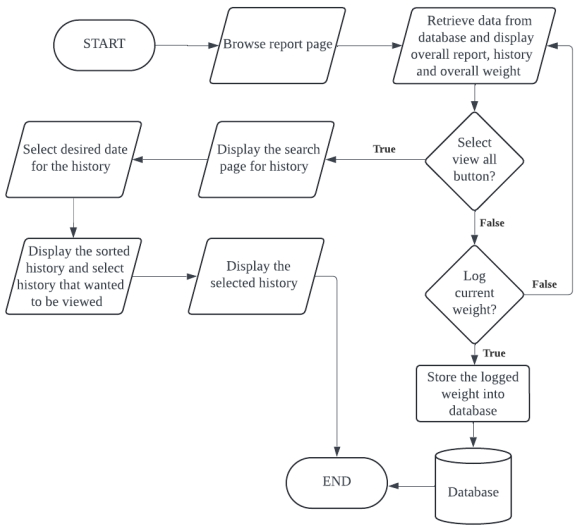


Fig. 20. flow chart for report module.

C. Algorithm Design

The algorithmic framework of this project is based on Google's ML Kit Pose Detection, a machine learning-based tool capable of identifying up to 33 anatomical landmarks in the human body. These landmarks encompass key joints and reference points, including the nose, eyes, shoulders, elbows, wrists, hips, knees, and ankles. By processing image or video input in real time, the pose detection model generates accurate spatial coordinates for each landmark, enabling detailed analysis of the user's posture and body orientation. This capability forms the basis for providing real-time feedback on exercise form and alignment.

This project uses a targeted subset of these anatomical landmarks to implement core motion capture functionalities, specifically tailored for the biomechanical analysis of squats and deadlifts. The subsequent sections detail the custom algorithms developed to process these landmarks, encompassing joint coordinate extraction, full-body visibility, joint angle calculation, repetition counting, posture error detection, and emergency fall detection. Each algorithmic component plays a critical role in enabling precise form evaluation, improving user safety, and delivering meaningful, real-time feedback during strength training exercises.

1) Extracting joint coordinates: The initial phase of the algorithm involves detecting and extracting key joint landmarks from the user's body using the Google ML Kit pose detection framework. Although the pose detection model identifies up to 33 anatomical landmarks, this system focusses on a critical subset, specifically shoulders, elbows, hips, knees, and ankles, which are most relevant for analyzing squat and deadlift movements. The landmarks index used in this project is: landmark 0 (for detecting whole body within the frame or not and sudden fall); landmarks 11, 12, 13, 14, 23, 24, 25, 26, 27, 28, 31, 32 (Fig. 4). The extracted coordinates serve as the foundation for subsequent calculations, including joint angle computation, positional tracking, and posture classification. Additionally, the algorithm incorporates a visibility check to ensure that all required joints are detectable in the frame, thus validating full-body visibility before proceeding with further analysis. Fig. 21 illustrates the pseudocode implemented for joint coordinate extraction.

```
FOR each frame:
  pose = runPoseDetection(frame)
  IF pose is null THEN
    SKIP frame
  END IF

  EXTRACT key joint coordinates:
    leftShoulder = pose[LEFT_SHOULDER]
    rightShoulder = pose[RIGHT_SHOULDER]
    leftHip = pose[LEFT_HIP]
    rightHip = pose[RIGHT_HIP]
    leftKnee = pose[LEFT_KNEE]
    rightKnee = pose[RIGHT_KNEE]
    etc.

  IF any key joint is null THEN
    DISPLAY "User not fully in frame"
    CONTINUE
  END IF
```

Fig. 21. Pseudocode used to extract joint coordinates.

2) Detect full body within camera frame: To ensure accurate detection and tracking, the system verifies the visibility of all critical joints. If one or more of these essential landmarks are not detected in each frame, the algorithm assumes that the user is not fully within the camera's view. In such cases, the system bypasses further analysis for that frame to avoid inaccurate posture evaluation and maintain the reliability of the feedback mechanisms (Fig. 22).

```
CHECK full-body detection:
  REQUIRED_JOINTS = [NOSE, LEFT_HIP, RIGHT_HIP,
LEFT_ANKLE, RIGHT_ANKLE]
  FOR joint IN REQUIRED_JOINTS:
    IF joint is missing or confidence < threshold:
      DISPLAY "Please align yourself in the frame"
      RETURN
```

Fig. 22. Pseudocode to detect and ensure user's body is within the frame.

3) Calculate angles between joints: Using the extracted coordinates, the system calculates joint angles, specifically at the knees and hips, to evaluate movement phases and assess posture. These angles are computed using a standard geometric method involving three points, where the joint of interest (e.g.

the hip) is positioned between two adjacent landmarks (e.g. the shoulder and the knee). By continuously analysing these angle values, the system can determine the depth of a squat, detect misalignments, and identify potential deviations from proper form (Fig. 23).

```
FUNCTION calculateAngle(A, B, C):
  angle = arccos( (AB² + BC² - AC²) / (2 * AB * BC) )
  RETURN angle

hipAngle = calculateAngle(shoulder, hip, knee)
kneeAngle = calculateAngle(hip, knee, ankle)
```

Fig. 23. Pseudocode designed to calculate the angles between joints.

4) Count repetitions (squat/deadlift): The repetition counting logic is based on tracking the sequential movement patterns of key joints, particularly the hips and knees, over time, as shown in Fig. 24. For squats, the system continuously monitors changes in hip and knee joint angles to identify the different phases of the exercise. Repetition is initiated when both the hip and knee angles decrease beyond a predefined threshold, indicating that the user is descending into the squat. The repetition concludes when these angles increase as the user returns to a standing position. To ensure accuracy, repetition is counted only when this full motion cycle is completed and specific posture conditions are met, such as appropriate squat depth and joint alignment.

```
INITIALIZE:
  phase = "Standing"
  successfulRepCount = 0
  failedRepCount = 0

ON EACH FRAME:
  UPDATE kneeAngle, hipAngle, handHeight
  UPDATE timestamp for motion duration

  // Squat Detection Logic
  IF motionType == "SQUAT":
    IF kneeAngle < 100 AND hipAngle < 100:
      IF phase == "Standing":
        phase = "Squatting"

    ELSE IF kneeAngle > 150 AND hipAngle > 150:
      IF phase == "Squatting":
        IF postureIsValid():
          successfulRepCount += 1
        ELSE:
          failedRepCount += 1
        END IF
        phase = "Standing"

  // Deadlift Detection Logic
  IF motionType == "DEADLIFT":
    IF handHeight below knee:
      IF phase == "Standing":
        phase = "Lowering"

    ELSE IF handHeight above hip:
      IF phase == "Lowering":
        IF postureIsValid():
          successfulRepCount += 1
        ELSE:
          failedRepCount += 1
        END IF
        phase = "Standing"
```

Fig. 24. Pseudocode used in this project to count repetitions.

For deadlifts, the system uses the relative position of the hands in relation to the hips and knees to identify the lifting and lowering phases. Repetition is detected when the hands descend below the level of the knees and then ascend back above the hips in a controlled, continuous motion. This movement must also meet predefined posture conditions, such as proper spinal alignment and full extension to the top—for a repetition to be considered valid. The use of relative joint positioning ensures that only complete and correctly executed repetitions are counted.

5) Detect incorrect reps (posture validation): The app continuously tracks key joint landmarks (such as the hips, knees, shoulders, and hands) across frames during a squat or deadlift. To validate the posture for each repetition, the system compares the relative positions of corresponding joints (e.g. left and right shoulders, hips, elbows, and hands) to ensure that the user is performing the movement in a controlled and balanced manner. For instance, if the hands are misaligned or one shoulder is significantly lower than the other, the system flags these discrepancies as errors. Error flags are assigned to each joint segment and, if any of these flags are activated, the repetition is deemed invalid. When an error is detected, the app displays a relevant error message on the screen (e.g. "Hands are not parallel") and excludes the repetition from the count (Fig. 25).

```
Initialize errorFlags = {
  handPosition: false,
  shoulder: false,
  hip: false,
  feet: false,
  elbowAngle: false
}

FOR each frame during motion capture:
  IF absolute(leftHand.y - rightHand.y) > hand_threshold:
    errorFlags.handPosition = true
    Show error message: "Hands are not parallel"

  IF absolute(leftShoulder.y - rightShoulder.y) > shoulder_threshold:
    errorFlags.shoulder = true
    Show error message: "Shoulders not aligned"

  IF absolute(leftHip.y - rightHip.y) > hip_threshold:
    errorFlags.hip = true
    Show error message: "Hips not aligned"

  IF absolute(leftFoot.y - rightFoot.y) > feet_threshold:
    errorFlags.feet = true
    Show error message: "Feet not aligned"

  IF elbow angle not within correct range (deadlift only):
    errorFlags.elbowAngle = true
    Show error message: "Elbow angle incorrect"

AT end of repetition:
  IF any errorFlags is true:
    Mark rep as incorrect
    Increase failed rep counter
  ELSE:
    Count rep as successful
```

Fig. 25. Pseucode used in this project to detect errors.

6) Emergency mechanism detection: The emergency mechanism within the motion capture system is designed to

detect sudden falls or unsafe postural deviations by analysing real-time body landmark data, as shown in Fig. 26. Specifically, the system calculates the trunk angle by measuring the slope between the midpoint of the user's shoulders and ankles. If this angle exceeds a predefined threshold (typically greater than 60 degrees) and experiences a rapid change between consecutive frames (e.g. more than a 25-degree shift), the system flags the event as a potential fall. To reduce the likelihood of false alarms, the angle difference is smoothed using exponential filtering, and a cooling counter is employed to debounce temporary fluctuations, ensuring that only genuine postural changes trigger an alert.

```
function detectEmergency(pose):
    if key landmarks (nose, shoulders, hips, ankles) are missing:
        return false

    upperMid = midpoint(leftShoulder, rightShoulder)
    lowerMid = midpoint(leftAnkle, rightAnkle)
    bodyAngle = angleBetween(upperMid, lowerMid)

    if lastAngle == 0:
        set lastAngle = bodyAngle
        return false

    angleChange = abs(bodyAngle - lastAngle)
    set lastAngle = bodyAngle

    if bodyAngle > 60 and angleChange > 25:
        increment dropFrameCount
        set emergency = true
        return true
    else:
        decrement dropFrameCount
        return false
```

Fig. 26. Pseudocode used to detect emergencies / injuries.

IV. Result

This section presents the results of the User Acceptance Testing (UAT) and accuracy evaluation conducted on the mobile applications Real-time Biomechanical Squat and Deadlift Posture Analysis, based on data collected from two distinct user groups.

A. Usability Test

The usability test (UT) was conducted with six experienced bodybuilders to assess the usability based on real-world user interactions (Fig. 27). Participants completed a full cycle of tasks, including account registration, profile setup, workout planning, motion capture, emergency alert testing, and performance reporting. The demographic profiles of the participants are summarised in Table II.

TABLE II. UAT Participants' Demographics

#	Age	Gender	Years of Bodybuilding	Beginner/ intermediate/ advance level	Occupation
1	22	Male	1	Beginner	Undergraduate
2	22	Male	1	Beginner	Undergraduate
3	22	Male	0.5	Beginner	Undergraduate
4	22	Male	0.5	Beginner	Undergraduate
5	22	Male	0.5	Beginner	Undergraduate
6	22	Male	3	Intermediate	Undergraduate

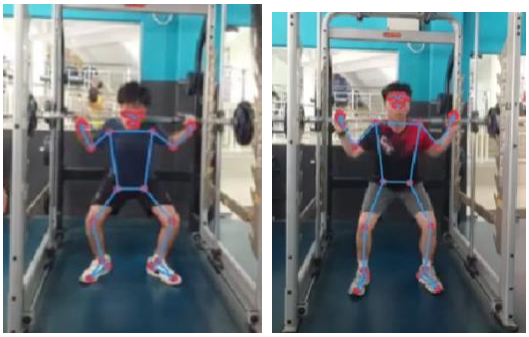


Fig. 27. Screenshots of usability test.

Feedback was collected through a structured questionnaire administered through Google Forms, consisting of 11 items rated on a 5-point Likert scale, alongside open-ended response sections (Fig. 28). In general, the usability of the application received positive evaluations. Specifically, 83.3% of the participants rated the user interface as easy to navigate (a score of 4), while 16.7% awarded it the highest score of 5. The workout planning feature was particularly well received, with 83.3% of users assigning it a score of 5, citing its simplicity and clarity.

UAT1. The app interface is easy to navigate. [Strongly disagree(1) – Strongly agree(5)]
UAT2. The workout planner is simple to use and understand. [Strongly disagree(1) – Strongly agree(5)]
UAT3. The motion capture detected my posture correctly. [Strongly disagree(1) – Strongly agree(5)]
UAT4. The real-time feedback during my squat/deadlift was clear and useful. [Strongly disagree(1) – Strongly agree(5)]
UAT5. The sound alert and emergency contact feature make me feel safer while working out. [Strongly disagree(1) – Strongly agree(5)]
UAT6. The graphs and analytics helped me understand my workout patterns. [Strongly disagree(1) – Strongly agree(5)]
UAT7. The weight log is useful to track my fitness progress over time. [Strongly disagree(1) – Strongly agree(5)]
UAT8. What did you like the most about the app? [Open-ended question]
UAT9. What part of the app did you find confusing or difficult to use? [Open-ended question]
UAT10. Do you have any suggestions for improvement? [Open-ended question]

Fig. 28. Questionnaires used in UAT.

On the contrary, the posture detection module –powered by Google's ML Kit –received moderate feedback. The majority of the participants (66.7%) rated their performance as average (score of 3), while the remaining 33.3% assigned it a slightly higher rating of 4 (Table III). These results indicate that, although the posture detection functionality is operational, its accuracy requires further optimisation to meet user expectations.

TABLE III. Responses of Participants in UAT1-7

UAT Questionnaire	Rating-Percentage in %	Average ratings
UAT1	4-83.3%; 5-16.7%	4.17
UAT2	4-16.7%; 5-83.3%	4.83
UAT3	3-66.7%; 4-33.3%	3.33
UAT4	3-33.3%; 4-33.3%; 5-33.3%	4.00
UAT5	4-50%; 5-50%	4.83
UAT6	3-16.7%; 4-50%; 5-33.3%	3.33
UAT7	3-16.7%; 33.3%; 5-50%	4

Similarly, user responses to the real-time exercise feedback feature were evenly distributed between ratings of 3, 4, and 5 (each representing 33.3%), reflecting a varied user experience. This inconsistency may stem from external factors, such as suboptimal camera angles or insufficient lighting conditions during motion capture, which can impact the effectiveness of real-time feedback.

In particular, all participants agreed that the application effectively helped them assess the correctness of their exercise form. Specifically, 50% rated this feature a 4, while the remaining 50% awarded it a 5, indicating strong user confidence in the form recognition capability. The emergency alert system, which integrates a loud alarm and automated contact notification, also received favourable reviews, with 83.3% of users assigning it a rating of 4 or 5.

Additional features, such as analytics and weight tracking, were similarly well received. The weight logging functionality was rated 4 or higher by 83.3% of participants, and an equal percentage reported that the data visualisation through workout charts enhanced their understanding of training patterns. Average UAT1-UAT7 ratings.

Open feedback offered additional information on both the strengths and potential areas for improvement within the application (see Table IV). The participants highlighted several appreciated features, including real-time posture correction, personalised weight recommendations, the integrated emergency alert system, and the intuitive reporting interface.

TABLE IV. PARTICIPANTS' RESPONSES FOR UAT8-10

UAT8: What did you like the most about the app?
1. The real-time feedback during squats was very helpful — it felt like having a trainer watching me.
2. I liked the emergency feature. It gave me peace of mind knowing it would alert someone if I needed help.
3. The design is clean and it was easy to log my weight and see the progress in the report.
4. I liked how it gives personalized workout recommendations based on fitness level.
5. It's cool how the camera can detect my posture and give feedback instantly.
6. Everything is in one place — workout planner, progress, and form correction.
UAT9: What part of the app did you find confusing or difficult to use?
1. The workout planner interface was a bit overwhelming at first.
2. I wasn't sure how to trigger the motion capture — had to press a few buttons to figure it out.
3. The emergency contact setup took me a while to find in the profile section.
4. The report charts were a bit small on my phone screen.
5. Sometimes I wasn't sure if the camera was properly detecting me or not.
6. Forgot password process was a bit long, especially the OTP step.
UAT10: Do you have any suggestions for improvement?
1. Maybe add a short tutorial for new users when they first open the app.
2. Consider adding labels or tooltips for each button to explain their function.
3. Maybe group the profile and emergency settings under a clearer menu.
4. Allow the charts to be expanded or rotated for better visibility.
5. Maybe show an indicator or guide when the app is actively tracking your movement.
6. Maybe allow the app to resend OTP without waiting too long.

However, users also provided constructive suggestions to improve the overall experience. These included enabling faster OTP (One-Time Password) resends, improving the organisation of profile and emergency settings, expanding report charts for better readability, and incorporating brief tutorials or tooltip explanations to support first-time users. Additionally, some participants recommended the inclusion of a visual indicator to confirm when motion tracking is actively engaged.

In summary, all participants successfully completed the assigned tasks and provided valuable constructive feedback. The application was deemed functionally complete, user-friendly and effective in achieving its intended purpose, thus affirming alignment with the original design objectives. While the overall user experience was positive, incorporating minor usability improvements based on user feedback could further refine the system and strengthen its readiness for a wider deployment.

B. Motion Capture Accuracy Assessment

To evaluate the accuracy of motion capture functionality within the Pose Perfect application, a focused assessment was conducted involving three students from the Faculty of Sport Science at Tunku Abdul Rahman University of Management and Technology (see Table V). Each participant had more than three years of practical experience with squat and deadlift exercises. They were invited to independently explore the application and provide feedback through a structured Google Form questionnaire, which evaluated the key posture-related features in terms of precision, responsiveness, and reliability. The details of the questionnaire items are presented in Fig. 29.

TABLE V. ACCURACY ASSESSMENT PARTICIPANTS' DEMOGRAPHIC

#	Age	Gender	Years of Bodybuilding	Beginner/ intermediate/ advance level	Occupation
1	22	Male	3	Intermediate	Undergraduate
2	22	Male	4	Intermediate	Undergraduate
3	22	Male	4	Intermediate	Undergraduate

ACC1. How accurately does the app detect your squat posture in real time? **[Not accurate (1) – Very accurate (5)]**
ACC2. How accurately does the app detect your deadlift posture in real time? **[Not accurate (1) – Very accurate (5)]**
ACC3. Does the app correctly identify incorrect posture during squats (e.g., unbalanced hands, knees not aligned)? **[Never (1) – Always (5)]**
ACC4. Does the app correctly identify incorrect posture during deadlifts (e.g., unbalanced hands, hands not straight)? **[Never (1) – Always (5)]]**
ACC5. How consistent is the real-time feedback timing? **[Very inconsistent (1) – Very consistent (5)]**
ACC6. Have you tested the emergency mechanism (e.g., simulating a fall or sudden collapse)? **[No/Yes]**
ACC7. If yes, how accurately did the app detect the emergency condition? **[Not at all (1) – Very accurately (5)]**
ACC8. Was the emergency alert (alarm + options) triggered at the correct moment during the fall simulation? **[No/Yes]**
ACC9. Would you recommend this app to users at the following fitness levels? **Beginner (No/Yes)**
ACC10 Would you recommend this app to users at the following fitness levels? **Intermediate (No/Yes)**
ACC11 Would you recommend this app to users at the following fitness levels? **Advanced (No/Yes)**
ACC12. Please rate the overall usefulness of the motion capture feature in assisting with safe and effective training: **[Very poor (1) – Excellent (5)]**

Fig. 29. Questionnaire items used in the accuracy test.

The results of the accuracy evaluation indicated that the application generally performed well in detecting squat posture in real time, and most participants rating its accuracy between moderate and high (Table VI). On the contrary, the detection of deadlift posture received slightly lower scores, with participant responses distributed more evenly within the moderate accuracy range. Regarding incorrect posture identification, participants acknowledged that the application could detect noticeable deviations; however, they also noted a need for improvement in recognising more subtle form discrepancies. The feedback on the responsiveness of the system's real-time guidance mechanism was largely positive, with the consistency and timing of feedback highlighted as a particular strength. All respondents agreed that this application is suitable for the beginner level but not for advanced level. Only one respondent would recommend this application to intermediate level users.

TABLE VI. RESPONDANTS' RATING ON ACCURACY

Accuracy (ACC) Questionnaire	Rating-Percentage in %	Average ratings
ACC1	2-33.3%; 4-66.7%	3.33
ACC2	2-33.3%; 3-33.3%; 4-33.3%	3.00
ACC3	3-66.7%; 4-33.3%	3.33
ACC4	3-66.7%; 4-33.3%	3.33
ACC5	4-66.7%; 5-33.3%	3.00
ACC6	YES-100%	-
ACC7	1-33.3%; 2-33.3%; 3-33.3%	3.33
ACC8	YES-66.7%; NO-33.3%	-
ACC9	YES-100%	-
ACC10	YES-33.3%; NO-66.7%	-
ACC11	NO-100%	-
ACC12	2-33.3%; 3-33.3%; 4-33.3%	3

In terms of safety features, all participants were able to successfully simulate emergency scenarios, such as a sudden fall, and confirmed that the emergency mechanism was activated as intended. However, the accuracy of detection varied, with most responses indicating only partial success. This suggests that, while the system is generally responsive under standard emergency conditions, its ability to recognise more nuanced or subtle incidents could be further optimised.

When evaluating the usefulness across different user experience, participants recommended it primarily for beginners, noting that its current functionality may be too limited for more advanced athletes. In general, the motion capture component was rated moderately effective in supporting safe and efficient training practices.

These findings underscore both the potential and limitations of the application. Although it offers a valuable foundation for form monitoring and injury prevention—particularly for novice users—enhancements in detection precision and feature adaptability could significantly broaden its applicability and effectiveness across a wider spectrum of users.

V. DISCUSSION

The project successfully achieved its defined objectives and was completed within the planned scope. The first objective, to examine existing applications in the market, was addressed through a comprehensive literature review and product comparison. This analysis identified key gaps, particularly in the areas of real-time feedback and user safety features, providing a strong rationale for the development of the proposed system.

The second objective, to implement real-time motion capture and feedback, was realised through Google's ML Kit for pose detection. This enabled the system to analyse users' body posture during squat and deadlift exercises and provide immediate visual cues to assist with the corrective action in real time. The third objective, to provide detailed performance analysis, was met by incorporating features to track and reporting exercise metrics, including repetition counts, posture accuracy, and weight logs. These insights allow users to monitor their progress over time and support ongoing self-improvement.

All major components outlined in the scope of the project were implemented, including user registration and registration, profile and settings management, workout planning, motion capture, emergency response functionality, and performance reporting. The system was extensively tested at multiple levels, and user acceptance feedback confirmed that the application met its core functional and usability requirements.

Among the key strengths is the effective implementation of real-time posture feedback using Google's ML Kit, which allows immediate and actionable guidance, particularly beneficial for user training without supervision. The emergency alert mechanism represents another significant achievement, enhancing user safety by detecting abrupt or unsafe motion (e.g. falls), triggering an alarm, and facilitating quick access to emergency contacts. Furthermore, by using Google's pre-trained ML Kit rather than developing a custom model, the project significantly reduced implementation complexity while maintaining satisfactory performance. Basic security measures, including password hashing and session management, were also implemented to ensure user data protection.

As compared to existing studies, Nguyen (2024) research detects static body pose from image and real-time yoga pose recognition. RepDetect [35] on the other hand, monitors exercises such as push-ups, sit-ups, etc. Groovetime and NeckFit provide real-time coaching in dance and body posture [21, 22]. However, gym workout is not included in the research. Existing market products such as Gym Log and Muscle Booster Plan Workouts mainly focus on providing exercises and digital planner without real time coaching.

Despite these achievements, several limitations were identified. The application was developed and tested exclusively on Android, limiting its accessibility across platforms. Additionally, the accuracy of the motion capture functionality is sensitive to environmental conditions such as lighting and camera angle, due to its reliance on visual input. Future improvements could involve the integration of device sensors or depth-sensing technologies to improve detection reliability. The current posture analysis is also based on fixed-angle rules, without adaptation to individual user variability or training history. Furthermore, time constraints limit the implementation of additional features such as workout reminders, voice guidance, and cloud-based data storage.

VI. Conclusion

The proposed system offers a creative and innovative solution to a real and often under-addressed challenge in the fitness technology domain: ensuring proper exercise form and workout safety during strength training without the need for direct supervision. Although many existing fitness applications prioritise features such as workout libraries, activity tracking, and goal setting, this project distinguishes itself by integrating real-time motion capture and posture analysis through Google's ML Kit. By focussing specifically on squats and deadlifts, two high-risk compound movements, the application provides users with immediate visual feedback, allowing them to correct their form in real time and reduce the risk of injury.

One of the most novel contributions is the built-in emergency mechanism, which identifies abnormal movement patterns, such as a fall or dangerous posture, and responds by triggering an audible alarm and initiating contact with a predefined emergency number. This safety feature addresses critical oversight in most fitness applications and adds considerable value for users who train independently, particularly at home or unsupervised gym settings.

In addition to its safety features, the system offers comprehensive tracking of exercise performance, including repetition counts, posture feedback, and progress monitoring through visual logs and charts. These elements enable users to take greater control of their fitness journey, allowing more informed self-assessment and long-term improvement.

The necessity of such a system is underscored by the fact that improper form, particularly during exercises such as squats and deadlifts, can result in serious injury, especially for beginners or users without professional supervision. By providing visual guidance during exercise execution, the application promotes safer training practices and supports proper biomechanics. Additionally, its mobile-based implementation enhances accessibility and convenience, eliminating the need for costly motion capture hardware or personal trainers.

From a marketability perspective, the application demonstrates strong potential. Its focused functionality, user-friendly design, and emphasis on safety align well with the growing demand for intelligent fitness tools. Positioned as a niche solution for strength training, it fills a gap in the current fitness app landscape by combining real-time posture correction with emergency safety features, two elements rarely found together in existing solutions.

A. Limitations and Future Improvements

Although the project successfully met its primary objectives, several limitations were identified during the development and testing phases.

A significant constraint is the current compatibility with only Android devices. This platform-specific implementation limits accessibility for iOS users and, therefore, reduces the potential reach of the application. Broadening the support of the platform is essential to maximize user adoption and market presence.

Another key limitation lies in the motion detection architecture, which relies solely on 2D pose estimation via a single camera angle. This configuration restricts accurate posture recognition to frontal views. If the user is positioned at an angle or partially outside of the frame, the accuracy of detection can be considerably degraded. Furthermore, the pose estimation model used in the application is optimised for larger joints and body segments and is less effective at detecting fine-grained joint movements. This limitation reduces the precision and depth of feedback, which is particularly important for advanced users or nuanced form corrections.

Functionality is further limited by the current scope of exercise support, which includes only squats and deadlifts. While these are foundational compound movements, the lack of support for additional exercises may hinder adoption by users with more diverse training programmes. Expanding the range of supported movements would make the system more versatile and appeal to a wider user base.

Language support presents another accessibility barrier. At present, the application is available only in English, which may exclude non-English speaking users and limit the app's appeal in international markets. Introducing multilingual support would improve usability and broaden the application's global relevance of the application.

Additionally, hardware limitations inherent to mobile devices present challenges to consistent performance. Real-time pose estimation requires continuous processing of video streams and joint calculations, which can strain lower-end devices. Users with limited processing power, low memory, or lower resolution cameras may experience reduced accuracy, frame rate drops, or latency, all of which impact the reliability of the posture feedback system. In high-precision use cases, desktop or high-performance environments may be more appropriate.

To address these limitations, several enhancements are recommended for future development. Expanding cross-platform compatibility to include iOS and desktop environments will enable greater accessibility. Improvements in the pose detection mechanism, such as incorporating multi-angle views or 3D tracking, would allow for more accurate recognition regardless of user orientation. Increasing the number of detectable joints and expanding the exercise library would enrich the system's functionality. Integrating multilingual support into the user interface will improve accessibility for a global audience. Finally, optimising the application for lower-end devices or offering a higher-performance version for PCs would improve consistency and detection reliability across different hardware configurations.

B. Issues and Solutions

One of the main technical challenges encountered during this project was the implementation of Google's ML Kit for pose detection in the context of real-time motion analysis for squats and deadlifts. As this was the first experience working with pose estimation technology, a substantial amount of time was invested in understanding how the model detects body landmarks and how the resulting coordinate data could be interpreted to assess posture. Beyond identifying joints, the system needed to logically evaluate whether a user was completing full repetitions, maintaining correct form, and avoiding potentially dangerous movements. This required designing algorithms capable of calculating joint angles,

tracking movement trajectories, and establishing thresholds to distinguish between proper and improper forms. Iterative design and testing of logic flows were necessary to ensure a reliable and consistent posture analysis mechanism.

Device performance presented another significant challenge during motion capture. The real-time detection process required continuous video processing and data analysis, which placed a heavy computational load on mobile devices. This led to overheating and accelerating battery depletion, particularly during extended exercise sessions. To address this issue, a rest mode feature was implemented. When a user completes a set, the system automatically pauses camera usage and suspends pose detection to allow the device to cool down. The user could resume the session manually using a "Continue" button. This approach helped mitigate device stress while maintaining session continuity.

In addition to technical hurdles, the project was also the first experience using the Flutter framework and the Dart programming language. Initially, the lack of familiarity with these tools introduced a steep learning curve, particularly in understanding the Flutter widget-based structure and state management principles. Early development progress was slower than expected due to this unfamiliarity and the increased difficulty of debugging. However, these challenges were progressively overcome through dedicated self-study, including tutorials, official documentation, and community forums, which eventually led to greater fluency and development efficiency.

Despite these obstacles, the challenges encountered throughout the project ultimately served as valuable learning experiences. The development process improved technical competencies in Flutter development, real-time motion detection, and logical algorithm design. As importantly, it fostered growth in project management, problem solving, and self-directed learning. Key takeaways included the ability to deconstruct complex problems into manageable units, adapt designs to accommodate real-world constraints, and implement continuous testing and refinement strategies to achieve reliable results. This is crucial as more advanced AI features could be integrated into the AI-powered fitness coach, such as cardiac detection to prevent fatal death and promote a more sustainable fitness programme to the society [36].

REFERENCES

[1] U.S. Army Public Health Center – Injury Prevention Program. Squat. 2020. https://ph.health.mil/PHC%20Resource%20Library/SquatInjuries_FS_12-025-1119.pdf.

[2] S. A. MAZZETTI et al., "The influence of direct supervision of resistance training on strength performance," M*edicine & Science in Sports & Exercise*, vol. 32, no. 6, pp. 1175–1184, Jun. 2000, doi: https://doi.org/10.1097/00005768-200006000-00023.

[3] S. S. Shim, J. E. Confino, and D. D. Vance, "Common Orthopaedic Injuries in CrossFit Athletes," *JAAOS - Journal of the American Academy of Orthopaedic Surgeons*, vol. 31, no. 11, p. 557, Jun. 2023, doi: https://doi.org/10.5435/JAAOS-D-22-01219.

[4] Mohsen Saffari et al., "Psychometric Properties of the Tendency to Avoid Physical Activity and Sport Scale Among Hong Kong University Students," Measurement in Physical Education and Exercise Science, pp. 1–11, Jan. 2024, doi: https://doi.org/10.1080/1091367x.2024.2302607.

[5] M. Menolotto, D.-S. Komaris, S. Tedesco, B. O'Flynn, and M. Walsh, "Motion Capture Technology in Industrial Applications: A Systematic Review," *Sensors*, vol. 20, no. 19, p. 5687, Oct. 2020, doi: https://doi.org/10.3390/s20195687.

[6] A. F. Panaite, S. Rosca, and R. Sibișanu, "Pose and motion capture technologies," *MATEC Web of Conferences*, vol. 342, p. 05004, 2021, doi: https://doi.org/10.1051/matecconf/202134205004.

[7] Iptif, "Motion Capture System," *IPTIF - IIT Palakkad Technology IHub Foundation*, Sep. 24, 2021. https://iptif.tech/motion-capture-system/.

[8] P. Rahimian and L. Toka, "Optical tracking in team sports," *Journal of Quantitative Analysis in Sports*, vol. 0, no. 0, Mar. 2022, doi: https://doi.org/10.1515/jqas-2020-0088..

[9] S. Wang and X. He, "The Development and Application of Computer Vision Technology in The Era of Artificial Intelligence," *Journal of Artificial Intelligence Practice*, vol. 5, no. 3, 2022, doi: https://doi.org/10.23977/jaip.2022.050301.

[10] S. García-de-Villa, D. Casillas-Pérez, A. Jiménez-Martín, and Juan Jesús García-Domínguez, "Inertial Sensors for Human Motion Analysis: A Comprehensive Review," *IEEE transactions on instrumentation and measurement*, vol. 72, pp. 1–39, Jan. 2023, doi: https://doi.org/10.1109/tim.2023.3276528.

[11] A. Barsegyan, "Perception Neuron Inertial Motion Capture vs Optical Mocap Systems and the First Production Motion Capture Session Experience - CGI coffee," *cgicoffee.com*. https://cgicoffee.com/blog/2017/04/first-production-motion-capture-session-report.

[12] A. Mathis, S. Schneider, J. Lauer, and M. W. Mathis, "A Primer on Motion Capture with Deep Learning: Principles, Pitfalls, and Perspectives," *Neuron*, vol. 108, no. 1, pp. 44–65, Oct. 2020, doi: https://doi.org/10.1016/j.neuron.2020.09.017.

[13] L. Jiang *et al.*, "Deep3DPose: Realtime Reconstruction of Arbitrarily Posed Human Bodies from Single RGB Images," *arXiv.org*, 2021. https://arxiv.org/abs/2106.11536 (accessed Sep. 12, 2025).

[14] E. Brau and H. Jiang, "3D Human Pose Estimation via Deep Learning from 2D Annotations," *2016 Fourth International Conference on 3D Vision (3DV)*, Stanford, CA, USA, 2016, pp. 582-591, doi: 10.1109/3DV.2016.84.

[15] A. Chatzitofis, A. Georgios, N. Zioulis, and S. Thermos, "Suit Up: AI MoCap," pp. 1–2, Jul. 2023, doi: https://doi.org/10.1145/3588430.3597249.

[16] Z. Lu, "Intelligent Processing Algorithms for Motion Capture Data Oriented to Artificial Intelligence," pp. 1–5, Feb. 2024, doi: https://doi.org/10.1109/icicacs60521.2024.10499186.

[17] K. Dimitropoulos, P. Daras, S. Manitsaris, F. Fol Leymarie, and S. Calinon, "Editorial: Artificial Intelligence and Human Movement in Industries and Creation," *Frontiers in Robotics and AI*, vol. 8, Jul. 2021, doi: https://doi.org/10.3389/frobt.2021.712521.

[18] C. Prins and S. Hu, "On-device machine learning solutions with ML Kit, now even easier to use," *Android Developers Blog*, 2020. https://android-developers.googleblog.com/2020/06/mlkit-on-device-machine-learning-solutions.html (accessed Sep. 12, 2025).

[19] "ML Kit," *Google Developers*. https://developers.google.com/ml-kit.

[20] C. Nguyen, "Utilizing Google's machine learning kit in developing Android applications ," Bachelor's Thesis, Metropolia University of Applied Sciences, 2024. Accessed: Sep. 13, 2025. [Online]. Available: https://www.theseus.fi/bitstream/handle/10024/858630/Nguyen_Chi.pdf.

[21] "Groovetime | Dance Challenge Central," *Groovetime.com*, 2025. https://www.groovetime.com/.

[22] B. Kitpitak, "Posture: NeckFit AI Workout," *App Store*, Sep. 11, 2024. https://apps.apple.com/th/app/posture-neckfit-ai-workout/id6670690205

[23] G. D. Myer *et al.*, "The Back Squat: A proposed assessment of functional deficits and technical factors that limit performance," *Strength and Conditioning Journal*, vol. 36, no. 6, pp. 4–27, Dec. 2014.

[24] M. A. Zulkifley, N. A. Mohamed, and N. H. Zulkifley, "Squat Angle Assessment Through Tracking Body Movements," *IEEE Access*, vol. 7, pp. 48635–48644, 2019, doi: https://doi.org/10.1109/access.2019.2910297.

[25] S. Zhang, W. Chen, C. Chen, and Y. Liu, "Human deep squat detection method based on MediaPipe combined with Yolov5 network," *2022 41st*

Chinese Control Conference (CCC), Jul. 2022, doi: https://doi.org/10.23919/ccc55666.2022.9902631.

[26] U. Dedhia, Pratham Bhoir, Prateek Ranka, and P. Kanani, "Pose Estimation and Virtual Gym Assistant Using MediaPipe and Machine Learning," Sep. 2023, doi: https://doi.org/10.1109/nmitcon58196.2023.10275938.

[27] "Dumbbell Bench Press Standards for Men and Women (lb)," *Strength Level*. https://strengthlevel.com/strength-standards/dumbbell-bench-press/lb.

[28] C. J. Holmes, "Understanding The Deadlift and Its Variations," *ACSM's Health & Fitness Journal*, vol. 24, no. 3, pp. 17–23, 2020, doi: https://doi.org/10.1249/fit.0000000000000570.

[29] J. Flandez, J. Gene-Morales, A. Juesas, Á. Saez-Berlanga, I. Miñana, and J. C. Colado, "A systematic review on the muscular activation on the lower limbs with five different variations of the deadlift exercise," *Journal of Human Sport and Exercise - 2020 - Summer Conferences of Sports Science*, 2020, doi: https://doi.org/10.14198/jhse.2020.15.proc4.27.

[30] S. Sutthiprapa, V. Vanijja, and T. Likitwon, "The deadlift form analysis system using Microsoft Kinect," *Procedia Computer Science*, vol. 111, pp. 174–182, 2017, doi: https://doi.org/10.1016/j.procs.2017.06.025.

[31] R. F. Escamilla, A. C. Francisco, A. V. Kayes, K. P. Speer and C. T. Moorman, "An electromyographic analysis of sumo and conventional style deadlifts," *Medicine & Science in Sports & Exercise*, vol. 34, no. 4, pp. 682–688, Apr. 2002, doi: https://doi.org/10.1249/00005768-200204000-00019.

[32] S. Lee, J. Schultz, J. Timgren, K. Staelgraeve, M. Miller, and Y. Liu, "An electromyographic and kinetic comparison of conventional and Romanian deadlifts," *Journal of Exercise Science & Fitness*, vol. 16, no. 3, pp. 87–93, Dec. 2018, doi: https://doi.org/10.1016/j.jesf.2018.08.001.

[33] "Gym Workout Tracker: Gym Log - Apps on Google Play," *play.google.com*. https://play.google.com/store/apps/details?id=gymworkout.gym.gymlog.gymtrainer&hl=en.

[34] WellTech Apps Limited, "Muscle Booster Workout Planner," *Google.com*, 2021. https://play.google.com/store/apps/details?id=musclebooster.workout.home.gym.abs.loseweight&hl=en (accessed Sep. 12, 2025).

[35] SurajKRB, "GitHub - SurajKRB/RepDetect: RepDetect is an android mobile application for workout enthusiast which uses Google MediaPipe Pose landmark detection using MLKit to create a basic fitness application.," *GitHub*, 2023. https://github.com/SurajKRB/RepDetect.

[36] K. Saranya, U. Karthikeyan, A. S. Kumar, Ayodeji Olalekan Salau, and T. T. Tin, "DenseNet-ABiLSTM: Revolutionizing Multiclass Arrhythmia Detection and Classification Using Hybrid Deep Learning Approach Leveraging PPG Signals," *International Journal of Computational Intelligence Systems*, vol. 18, no. 1, Feb. 2025, doi: https://doi.org/10.1007/s44196-025-00765-z.

A Novel CNN-Based Feature Fusion Framework for Breast Cancer Ultrasound Image Classification

Mobarak Zourhri[1], Bouchaib Cherradi[2], Mohamed El Khaili[3]
EEIS Laboratory-ENSET of Mohammedia, Hassan II University of Casablanca, Mohammedia 28830, Morocco[1, 2, 3]
STIE Team-CRMEF Casablanca-Settat, Provincial Section of El Jadida, El Jadida 24000, Morocco[2]

***Abstract*—Breast cancer remains a major global health concern and is among the leading causes of cancer-related deaths in women. Timely and precise diagnosis significantly improves treatment outcomes and patient survival rates. This paper presents a novel deep learning-based framework for breast cancer classification using ultrasound imagery, built upon the concatenation of two pre-trained Convolutional Neural Network (CNN) models: VGG19 and EfficientNetB0. By leveraging transfer learning and combining heterogeneous feature representations, the proposed method enhances the discriminative power of the extracted features. The model is evaluated on a publicly available benchmark ultrasound dataset and assessed through standard performance indicators, including accuracy, precision, recall, F1-score, and Area Under the Curve (AUC). In addition, Gradient-weighted Class Activation Mapping (Grad-CAM) is employed to generate interpretability heatmaps, visually highlighting regions that contribute most to classification outcomes. The experimental findings reveal that the integrated architecture outperforms several existing approaches as well as individual CNN baselines. This study contributes to the growing field of AI-assisted medical diagnostics and demonstrates the effectiveness of model fusion in ultrasound-based breast cancer detection.**

***Keywords*—Breast cancer classification; ultrasound imaging; Convolutional Neural Networks (CNN); transfer learning; model fusion; Grad-CAM; deep learning**

I. Introduction

Breast cancer continues to be one of the most frequently diagnosed and life-threatening diseases among women globally. As reported by the World Health Organization (WHO), over 2.3 million women were diagnosed with breast cancer in 2020, leading to approximately 685,000 deaths globally[1]. Early diagnosis remains a cornerstone in the fight against breast cancer, contribute significantly to lower mortality rates and enhancing the effectiveness of therapeutic interventions [1].

Ultrasound remains a commonly employed adjunct technique for breast cancer screening, particularly beneficial for women with dense breast tissue where mammography may fall short in sensitivity. Its widespread use is attributed to its non-invasive nature, cost-effectiveness, and ability to provide real-time imaging [2]. Nevertheless, the diagnostic reliability of ultrasound can be compromised by its operator-dependent interpretation, often resulting in variations between observers and inconsistent assessments [3].

In recent years, deep learning, especially Convolutional Neural Networks (CNNs), has gained considerable traction for automating the interpretation of medical images [4]. These models have demonstrated strong performance in classification tasks by learning discriminative features directly from data [5], thereby eliminating the need for manual feature engineering [6]. Nevertheless, constructing deep learning models specifically for medical applications poses considerable challenges, largely because of the limited availability of large-scale, high-quality annotated datasets in this domain.

To address the challenge of limited medical data, transfer learning has become an effective and widely utilized approach. By leveraging models pre-trained on large-scale datasets such as ImageNet, researchers can fine-tune neural networks for domain-specific medical imaging tasks [7]. In addition, combining different CNN architectures through ensemble methods or feature-level concatenation has been investigated as a means to enhance classification accuracy and generalization by capturing complementary feature representations [8]. In this context, several research areas have benefited from this development, such as: Cardiovascular disease classification [9], [10], [11], diabetes disease prediction [12], [13], [14], Parkinson's disease detection [15], [16], [17], handwritten recognition [18], [19], [20], sentiment analysis [21], [22], etc.

In this study, we introduce a dual-CNN fusion framework designed for the classification of breast cancer from ultrasound images. Our proposed approach combines the feature extraction capabilities of VGG19 and EfficientNetB0 through a feature-level concatenation strategy, while keeping the pre-trained convolutional blocks frozen to reduce overfitting and accelerate convergence. The integration of Gradient-weighted Class Activation Mapping (Grad-CAM) further enhances model interpretability by providing visual explanations of the regions that most influenced the classification outcome.

The effectiveness of the proposed model was evaluated using a publicly available breast ultrasound image dataset. The model achieved highly competitive performance with an accuracy of 98.44%, precision of 98.55%, recall of 99.66%, F1-score of 99.11%, and an AUC of ≥ 0.99. These results demonstrate the potential of our fusion-based approach to support early and accurate breast cancer diagnosis, offering a valuable contribution to computer-aided diagnostic systems in clinical settings.

[1]https://www.who.int/news-room/fact-sheets/detail/breast-cancer

The rest of this article is structured as follows: Section II reviews the most relevant contributions and previous research efforts related to deep learning applications for breast cancer diagnosis. In Section III, the proposed deep convolutional framework based on the integration of VGG19 and EfficientNetB0 is described in detail, enhancement the architectural choices and fusion strategy. Section IV outlines the dataset characteristics, preprocessing techniques, and evaluation metrics adopted in this study. Section V reports the experimental findings and performance metrics achieved by our model, including the aforementioned high scores confirming the strength of the proposed method. Finally, Section VI concludes the study and outlines potential directions for future work.

II. Related Works

Recent advances in artificial intelligence, particularly in deep learning, have substantially improved the classification and diagnosis of breast cancer using medical imaging. In past studies, traditional feature extraction and dimensionality reduction techniques were most commonly applied to image classification problems, such as the use of homogeneity features in combination with mutual information for hyperspectral image analysis [23]. Among these techniques, Convolutional Neural Networks (CNNs) have demonstrated remarkable success in extracting relevant and discriminative features from complex image data, especially in ultrasound imaging, which remains a widely used modality for breast cancer screening due to its non-invasive nature and cost-effectiveness.

Several studies have explored the integration of CNN architectures and transfer learning techniques for enhancing breast lesion classification. For instance, the authors in [24] developed a computer-aided diagnosis (CAD) system based on an ensemble of VGG19 and ResNet152 to distinguish between benign and malignant lesions in breast ultrasound images. Their model achieved a sensitivity of 90.9% and an Area Under the Curve (AUC) of 0.951, showcasing the effectiveness of combining deep architectures.

Similarly, in [25], the authors employed an ensemble learning strategy involving VGG, ResNet, and DenseNet networks to classify ultrasound images of breast tumors. Their evaluation on both public and private datasets resulted in an accuracy of 94.62%, a recall of 92.31%, and an F1-score of 91.14% on the BUSI dataset, indicating the benefits of leveraging multiple pre-trained networks to improve generalization and robustness.

The authors in [26] proposed a CNN-based approach utilizing GoogLeNet for binary classification of breast lesions. Their model achieved an accuracy of approximately 92.5% and a recall of 95.8%, highlighting the capability of deep learning to match or even surpass the diagnostic performance of experienced radiologists in certain clinical scenarios.

In another comparative study [27], the authors analyzed several CNN architectures, including VGG16 and InceptionV3, for breast tumor classification using ultrasound images. Using a dataset of 947 training images and 269 testing images, their best-performing model (fine-tuned VGG16) reached an accuracy of 91.9% and an AUC of 0.934, emphasizing the critical role of fine-tuning and architecture selection in CNN-based medical applications.

In our prior work [28], we conducted a comparative analysis of individual CNN models VGG16, VGG19, MobileNetV2, and ResNet50V2 using the same breast ultrasound dataset employed in this current study. Among them, the VGG19 model achieved the highest performance with an accuracy of 98.44%. Despite these encouraging results, single-network architectures are limited in capturing diverse feature representations and often lack interpretability, which is essential for clinical adoption.

To address these limitations, the present study proposes a dual CNN concatenation framework combining VGG19 and EfficientNetB0, selected for their complementary architectural properties and feature extraction capabilities. This model enhances performance with feature-level fusion without compromising generalizability by freezing pre-trained convolutional heads. Moreover, incorporating Grad-CAM visualization advances transparency by uncovering the most discriminative region in each input image, rendering the system more interpretable and reliable for clinical use.

III. Proposed CNN-Based Model

This section describes the architecture of the proposed deep learning model, which would classify breast ultrasound images as benign or malignant. The main goal is to improve diagnostic accuracy using transfer learning and feature combination via model concatenation.

A. System Overview

Fig. 1 illustrates the whole pipeline of the breast cancer diagnosis system. The system includes several steps starting from image acquisition and preprocessing of the ultrasound images, then feature extraction through pre-trained convolutional neural networks (CNNs). The features extracted from both models are combined and fed to a series of fully connected layers to generate the final binary prediction. This framework aims to be strong, understandable, and amenable to embedding within clinical workflow.

B. Model Architecture

The major contribution of this paper is the two-model concatenation method that combines two powerful pre-trained CNNs, VGG19 and EfficientNetB0, at the feature level. Both models are used as feature extractors without their classification heads, and only the last layers are fine-tuned to the ultrasound domain.

Each backbone processes the input image in parallel, generating high-level feature representations. The feature maps are concatenated through a concatenation layer. The fused feature vector is fed through a sequence of fully connected layers with dropout regularization to avoid over-fitting. Finally, a sigmoid activation function is used to calculate the probability of malignancy. Fig. 2 illustrates the architecture of the proposed convolutional neural network.

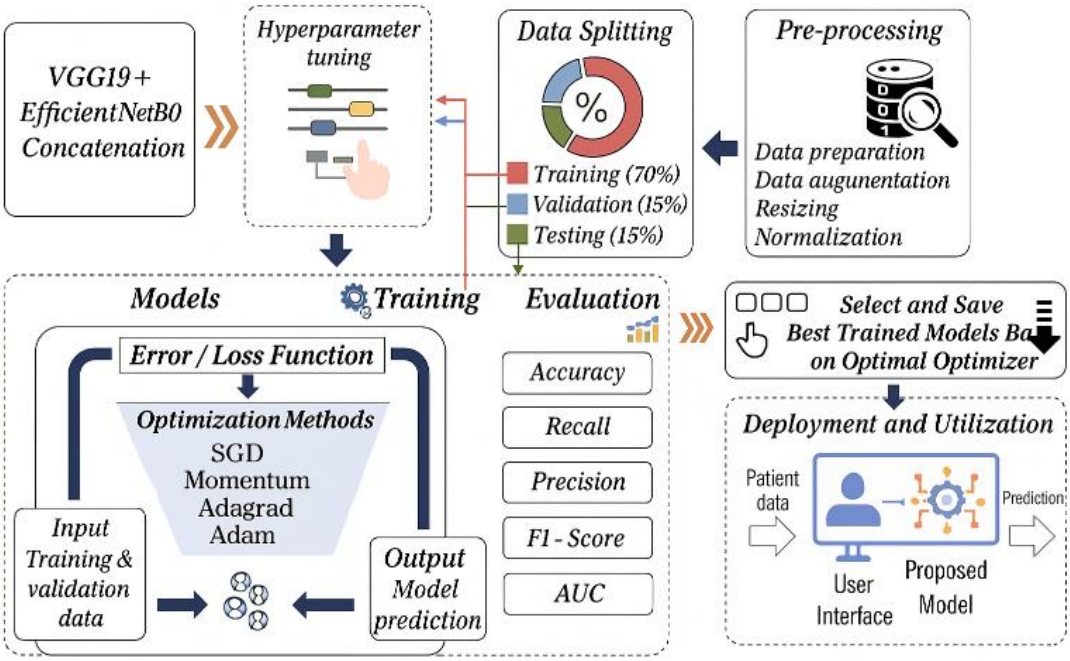


Fig. 1. Flowchart of the system for detecting breast cancer.

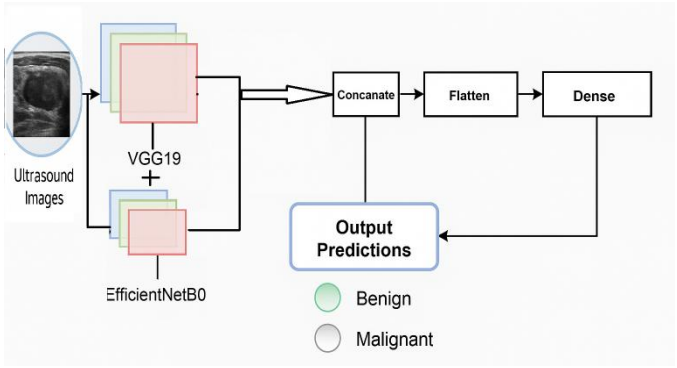


Fig. 2. The proposed CNN model architecture.

C. Motivation for Concatenation

The rationale behind using model concatenation is to harness the complementary strengths of both CNNs. VGG19 is known for its depth and ability to capture fine-grained spatial features, while EfficientNetB0 is optimized for efficiency and generalization. By combining their representations, the model can better distinguish subtle patterns present in benign and malignant lesions, which is particularly beneficial given the complexity of ultrasound data.

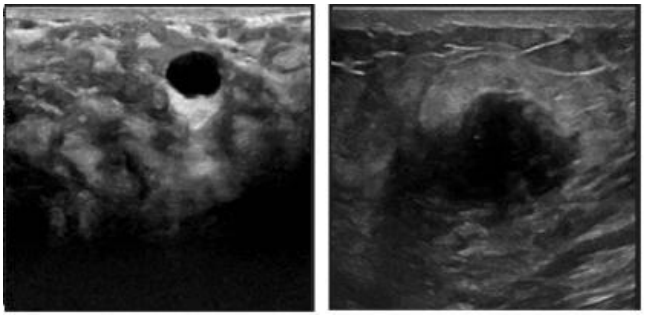
D. Output and Prediction

The output layer is a single neuron with a sigmoid activation function, producing a probability score between 0 and 1. A threshold of 0.5 is applied to categorize the image as benign or malignant. The performance of this architecture was thoroughly evaluated using various metrics, including accuracy, precision, recall, F1-score, and AUC.

IV. Materials and Methods

A. Dataset Collection

The dataset used in this study consists of breast ultrasound images categorized into two classes: benign and malignant. This dataset was previously employed in our earlier study [28] and is publicly available on Kaggle under the title "Ultrasound Breast Images for Breast Cancer"[2]. It comprises 9016 images in total, including 4574 benign and 4442 malignant samples. The images were preprocessed by resizing them to 224×224 pixels and normalizing pixel intensities to the [0,1] range. The choice of this dataset is motivated by its widespread use in prior research and its applicability to benchmarking breast cancer classification with ultrasound imaging, which remains one of the hardest modalities to handle since it is noisy and low-contrast Fig. 3 provides a visual representation of the sample data included in the database used in this paper.



(a) (b)

Fig. 3. Some samples of US images from the dataset are used: (a) Benign. (b) Malignant.

[2]https://www.kaggle.com/datasets/vuppalaadithyasairam/ultrasound-breast-images-for-breast-cancer

To ensure fair evaluation, the dataset was split into training (70%), validation (15%), and test (15%) subsets using stratified sampling to maintain class distribution. Table I summarizes the number of samples per class in each subset.

TABLE I. SAMPLE DISTRIBUTION ACROSS DATASET SPLITS

Subset	Benign	Malignant	Total
Training	3203	3109	6311
Validation	686	667	1353
Test	686	666	1352

B. Background on CNN

Convolutional Neural Networks (CNNs) are a class of deep learning models particularly effective in analyzing image data [29]. CNNs apply a series of convolutional and pooling layers to extract hierarchical features from input images. In this study, we employed a dual-branch CNN architecture combining VGG19 [30] and EfficientNetB0 [31], both pre-trained on the ImageNet dataset. The extracted features from both networks were concatenated before being passed through fully connected layers, enabling more comprehensive representation learning.

C. Confusion Matrix and Evaluation Metrics

In the context of binary classification for breast cancer diagnosis, a confusion matrix is an essential tool to assess the performance of the proposed model. It provides a detailed breakdown of the model's predictions compared to the actual ground truth labels by summarizing the number of correct and incorrect classifications.

The binary confusion matrix [32] is structured as a 2×2 table comprising the following components:

- True Positives (TP): Malignant cases correctly identified.
- True Negatives (TN): Benign cases correctly identified.
- False Positives (FP): Benign cases misclassified as malignant.
- False Negatives (FN): Malignant cases misclassified as benign.

These components form the foundation for calculating several widely adopted evaluation metrics that reflect different aspects of classification performance [33]:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{F1-Score} = 2 \times \frac{\text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \quad (4)$$

These are particularly critical in medical imaging use cases, where false negatives, specifically misclassification, can have profound implications on patient outcomes [34], [35]. They present a holistic perspective of the classification capacity of a deep learning algorithm and are critical indicators in determining whether it is ready for clinical adoption [36].

D. Gradient-Weighted Class Activation Mapping (Grad-CAM)

Gradient-weighted Class Activation Mapping (Grad-CAM) is a visualization technique used for the improved interpretability of convolutional neural networks (CNNs), especially in high-risk applications such as medical image processing. Grad-CAM generates class-specific localization heatmaps which display the most significant areas in an input image with respect to the prediction by the model, and thereby makes an insightful diagnosis of the decision-making process of the neural network.

This is particularly helpful in medical image analysis, where it is not only important to know the decision that a model made but also why a model made that decision. The technique allows localization of discriminative regions within an image by utilizing the gradient information flowing into the last convolutional layer.

The Grad-CAM method, proposed by [37], calculates the gradients of the class score with respect to the feature maps of the final convolutional layer. They are globally averaged to obtain importance weights, which are used to compute a weighted sum of the feature maps. The resulting localization map highlights the discriminative regions that considerably influence the model's prediction.

$$\alpha_k{}^c = \frac{1}{Z} \times \Sigma_i \sum_j \left(\frac{\partial y^c}{\partial A_{ij}{}^k} \right) \quad (5)$$

Where $\alpha_k{}^c$ represents the importance weight for feature map k with respect to class c, y^c is the score for class c, $A_{ij}{}^k$ denotes the activation at position (i, j) in the k-th feature map, and Z is the total number of pixels in A^k.

$$L^c_Grad - CAM = ReLU\ (\sum_k \alpha_k{}^c \times A^k) \quad (6)$$

This class-discriminative localization map, computed through the ReLU operation, specifically brings into view solely those features that have a positive contribution to the class of interest. This enables clinicians to better understand the model's decision-making process and ensures its alignment with medically relevant areas.

Grad-CAM was employed in this work on the presented concatenated CNN model to identify the discriminative regions responsible for classifying breast ultrasound images into benign and malignant classes. Grad-CAM identifies the spatial locations influencing the classification outcome by computing gradients of the target class with respect to the last convolutional layer. This proves useful in the medical context, where the explainability and transparency are critical towards clinical adoption [38].

The Grad-CAM technique enhances the transparency of the model to an extent that clinicians and radiologists are able to check the region of interest of the model and trust its predictions. Adding Grad-CAM to the framework not only increases diagnostic confidence but also facilitates the explanation of complex medical images.

Recent studies have emphasized the importance of explainable AI (XAI) techniques, such as Grad-CAM for

improving clinical applicability and regulatory compliance in healthcare AI solutions [39].

V. Experimental Results

A. Algorithm Best Parameters

In order to optimize the performance of the proposed deep learning model for breast cancer diagnosis from ultrasound images, various training settings and hyperparameters were chosen based on extensive experimentation. The training approach used was carried out within the TensorFlow Keras environment of a Kaggle notebook setup, which gave the computational capacity necessary for handling deep learning operations.

The input images were resized to a fixed resolution of 224 × 224 pixels, a size that is widely accepted to ensure compatibility with the input layer of most pre-trained CNN models, including VGG19 and EfficientNetB0. The data was divided into three sets: 70% for training, 15% for validation, and 15% for testing. This split allowed confident model estimation while still having sufficient data for training.

The training procedure was established with a batch size of 16 and a total of 24 epochs, which were empirically discovered to achieve an optimal trade-off between convergence rate and model generalization. The Adam optimizer [40] was utilized with learning rate set as 1e-5, enabling adaptive learning as well as stable training, particularly when fine-tuning the pre-trained networks.

To prevent overfitting and ensure generalizability, early stopping was employed with patience of 10 epochs, monitoring the validation loss as the primary stopping point. This ensured that training would be stopped once the performance of the model stabilized, preventing unnecessary computation and overfitting.

Besides, transfer learning was leveraged by loading the ImageNet pre-trained weights for VGG19 and EfficientNetB0. To train, the classification heads of both models were removed and only the shared feature maps were retained. Importantly, all convolutional layers in both networks were frozen to preserve the learned representations, except for the last classification block consisting of concatenated features and fully connected layers. This architecture, dubbed the "frozen heads" setup, takes advantage of powerful feature extractors while concentrating learning capacity on classification layers specific to the task.

To ensure maximum performance of the proposed concatenated model, a group of experiments was conducted to optimize the training configuration. Certain combinations of hyperparameters were investigated, and the choice was based on validation performance, model generalizability, and computational efficiency. The set of optimal hyperparameters utilized in the training of the model is depicted in Table II.

TABLE II. Optimal Hyperparameters Used for Training the Proposed Model

Network	Learning Rate	Batch Size	Optimizer	Loss Function	Epochs
The Proposed Model	1e-5	16	Adam	Binary Cross entropy	24

These parameters were set based on empirical results of repeated training iterations in the Kaggle Notebooks platform. The selected parameters gave a balance between achieving high accuracy and minimizing overfitting, especially when working with a relatively limited medical imaging dataset.

B. Training Results

During the training phase, the performance of the proposed model was monitored closely with standard parameters, including training loss, training accuracy, validation loss, and validation accuracy. All these parameters were recorded over 24 epochs to observe how the model learns and to detect evidence of overfitting or underfitting.

Fig. 4 and Fig. 5 present the accuracy and loss curves of training and validation datasets. In Fig. 4, training and validation accuracy grew with every epoch, which reflects the model to learn discriminative features from input ultrasound images. Fig. 5 exhibits the training and validation loss curves, which gradually came down over the period, reflecting convergence.

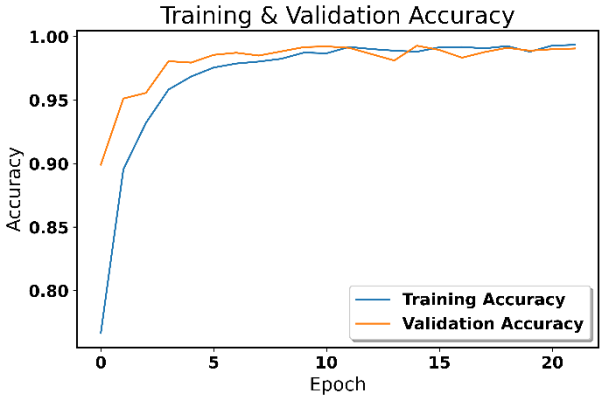


Fig. 4. Accuracy of the proposed model.

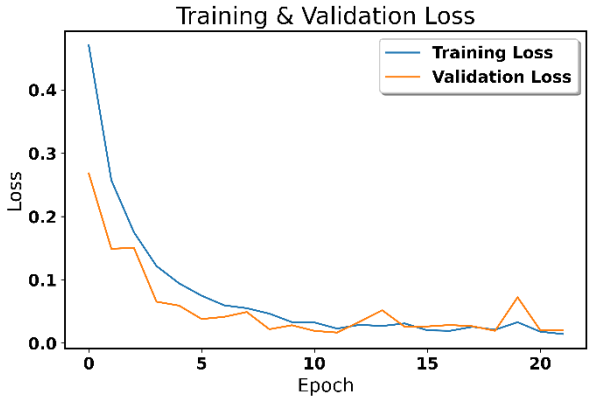


Fig. 5. Loss of the proposed model.

The training results indicate that the model generalizes well and is not overfitting, as the gap between the training and validation metrics remained minimal throughout the training process. Early stopping was employed to prevent overfitting, halting the training if no improvement was observed in the validation loss after 10 consecutive epochs.

These findings confirm the effectiveness of the proposed concatenated architecture in learning meaningful

representations from breast ultrasound images, contributing to robust classification performance.

C. Testing Results

To assess the generalization performance of the proposed model, the testing phase was conducted using the reserved 15% of the dataset. The model demonstrated strong predictive capabilities, as evidenced by the metrics summarized in Table III. These results highlight the model's effectiveness in correctly identifying benign and malignant breast tumors from ultrasound images.

TABLE III. TESTING PERFORMANCE METRICS OF THE PROPOSED MODEL

Metric	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC
Proposed Model (VGG19 + EfficientNetB0)	98.44	98.55	99.66	99.11	0.99

The evaluation metrics were computed from the confusion matrix and the ROC curve.

1) Confusion matrix: The confusion matrix shown in Fig. 6 summarizes the classification outcomes:

- True Positives (TP) = 886 malignant cases correctly predicted as malignant.
- True Negatives (TN) = 902 benign cases correctly predicted as benign.
- False Positives (FP) = 13 benign cases incorrectly classified as malignant.
- False Negatives (FN) = 3 malignant cases incorrectly classified as benign.

These results indicate a strong classification capability, especially with a very low false-negative rate, which is critical in clinical settings to avoid missing malignant cases.

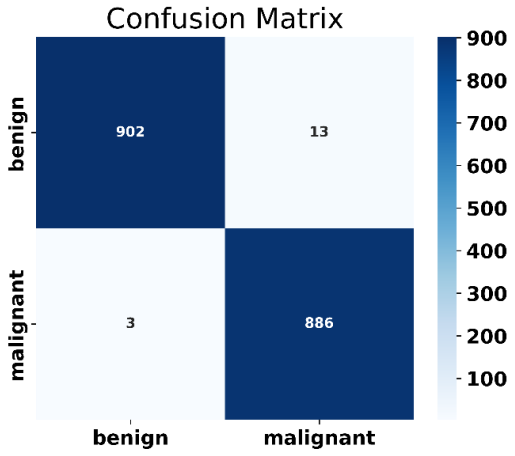


Fig. 6. The confusion matrix of the proposed model.

2) ROC curve and AUC: The ROC curve, depicted in Fig. 7, evaluates the model's discriminative power by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR). The Area Under the Curve (AUC) for the proposed model reached an impressive 0.9999, which signifies excellent reparability between the benign and malignant classes.

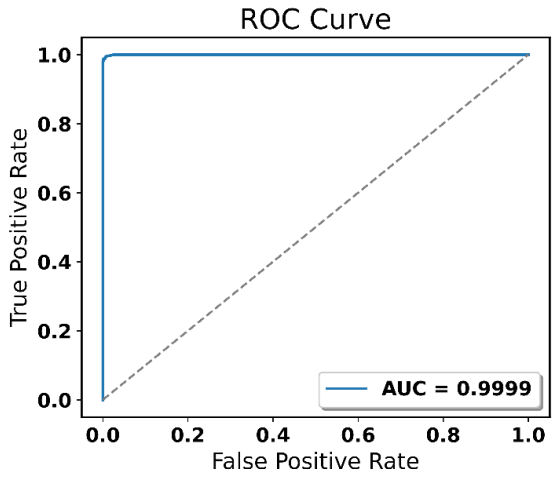


Fig. 7. ROC curve finding for the proposed model.

These findings confirm that the model not only achieves high accuracy (98.44%), but also maintains excellent precision (98.55%), recall (99.66%), and F1-score (99.11%), making it highly reliable for real-world breast cancer diagnosis using ultrasound imaging.

D. Discussion

The proposed dual-model CNN framework, based on the concatenation of VGG19 and EfficientNetB0, demonstrated notable improvements in breast cancer ultrasound image classification. This section discusses the significance of the obtained results, the visual interpretability provided by Grad-CAM, comparative performance with prior studies, limitations, and clinical implications. While the model was specially designed and optimized for breast ultrasound images, it is specifically well-suited for medical imaging modalities with similar issues of low contrast, noise, and high intra-class variability. However, the architecture can easily be generalized to other imaging applications, provided sufficient annotated data, which proves its potential generalizability to breast ultrasound.

1) Performance superiority of the proposed model: Our proposed model achieved an accuracy of 98.44%, a precision of 98.55%, a recall of 99.66%, an F1-score of 99.11%, and an AUC value of 0.9999. These accuracies beat those of individual backbone networks and most prior models reported in the literature. The complementarity of feature representations learned by EfficientNetB0 and VGG19 enabled the model to learn nuanced spatial and contextual patterns, resulting in a greatly improved classification performance.

The improved performance of the proposed model owes to the complementary capability of the backbone networks. VGG19 captures deep spatial features, and EfficientNetB0 offers parameter efficiency and enhanced generalizability.

Concatenation of both at the feature level increases the richness in the extracted features, enabling more precise classification.

2) Grad-CAM visualizations for model interpretability

a) Interpretation of the proposed model: To improve the interpretability of the proposed concatenated model (VGG19 + EfficientNetB0), Gradient-weighted Class Activation Mapping (Grad-CAM) was applied to visualize the discriminative regions that influence the model's classification decisions. Grad-CAM enables a qualitative assessment by projecting class-specific activation maps onto the original ultrasound images, thereby highlighting the areas the model focuses on when distinguishing between benign and malignant cases.

Fig. 8 shows an original breast ultrasound image of a benign lesion. The grayscale image serves as a reference, showing the anatomical structures without any model-generated overlay.



Fig. 8. Original breast ultrasound image showing a benign lesion.

Fig. 9 shows the Grad-CAM visualization generated by the new model. The heatmap overlays reveal that the model is paying attention most of the time to the lesion itself, with the most intense activations marked in red and yellow. This pattern of activation is in line with clinical expectations for benign lesions, where the model's attention is contained within the lesion pattern, aligning with clinical expectations for benign lesions, as the model's attention is confined within the lesion boundaries, avoiding unnecessary activation in adjacent healthy tissues.

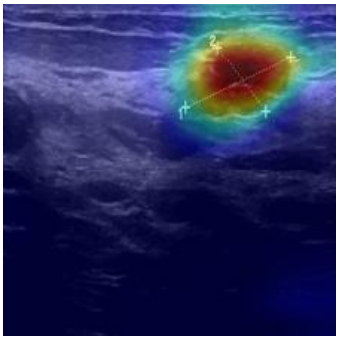


Fig. 9. Grad-CAM visualization from the proposed VGG19 + EfficientNetB0 model, emphasizing the lesion boundary influencing the benign classification decision.

The model's accuracy in targeting the lesion area means that the concatenated structure is capable of encoding the salient aspects of benign cases. This contributes significantly towards encouraging both the interpretability and clinical validity of its predictions. Moreover, the Grad-CAM visualizations positively establish that the model arrives at decisions on the basis of anatomically relevant areas, rather than being confused by noise or noise-like image artifacts.

This interpretability analysis places the model's clinical value into perspective by showing how accurately it can identify important diagnostic areas on ultrasound scans. Providing such visual feedback is crucial when integrating AI-based diagnostic systems into medical procedures, as it fosters transparency and credibility with clinicians who rely on these systems for well-educated choices.

b) Comparative Analysis with other CNN architectures: To further compare the interpretability analysis of the proposed dual-model architecture (VGG19 + EfficientNetB0), there was qualitative comparison against three top single standalone convolutional neural networks: VGG16, EfficientNetB0, and GoogleNet. Comparison was done for all the models using Grad-CAM heatmaps that were implemented on a corresponding benign breast ultrasound image, hence preserving consistency in comparative visual interpretation.

Fig. 10 shows the corresponding Grad-CAM heatmaps in 2×2 formation: top-left shows VGG16, top-right shows EfficientNetB0, bottom-left shows GoogleNet, and bottom-right shows the proposed model (VGG19 + EfficientNetB0).

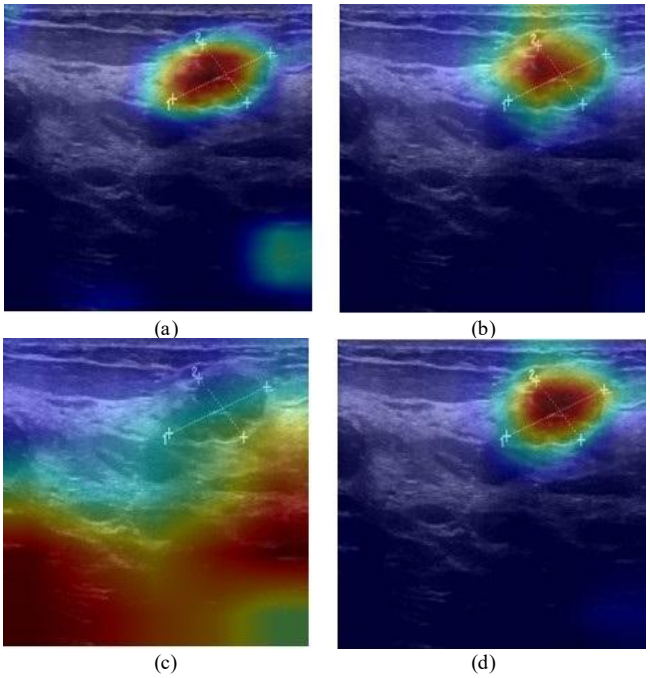


Fig. 10. Visualizations of benign lesion classification Grad-CAM comparison between different CNN architectures: (a) VGG16, (b) EfficientNetB0, (c) GoogleNet, (d) Proposed VGG19 + EfficientNetB0.

These heat maps represent the spatial attention of each model in processing the same benign case. As one may observe, the proposed concatenated model has the most concentrated activation within the diseased area, with minimal interference from the surrounding tissue, suggesting good clinical concordance.

In contrast, VGG16 and EfficientNetB0 show moderate attention to the periphery, while GoogleNet tends to produce more dispersed activation maps, sometimes extending beyond the boundaries of the lesion. These distinctions highlight the value of combining complementary feature representations in the proposed model.

To support the visual observations, a qualitative analysis was conducted based on four interpretability criteria: Activation Focus, Localization Precision, Background Noise, and Clinical Interpretability. The results are summarized in Table IV.

TABLE IV. QUALITATIVE COMPARISON OF GRAD-CAM INTERPRETABILITY ACROSS CNN MODELS

Model	Activation Focus	Localization Precision	Background Noise	Clinical Interpretability
VGG16	Moderate focus on lesion	Medium	Low to medium	Acceptable
EfficientNetB0	Peripheral focus	High at edges	Low	Good
GoogleNet	Scattered focus	Low	High	Limited
Proposed Model	**Centered on lesion**	**Very High**	**Very Low**	**Excellent**

This comparative analysis reinforces the superior interpretability and diagnostic accuracy of the proposed model. The combination of precise localization via heat maps and high classification metrics suggests that feature fusion via model concatenation can significantly improve the performance and clinical usability of AI-based diagnostic tools. This information is particularly relevant for real-world deployment, where model transparency and reliability are crucial factors for their integration into medical decision-making processes.

3) Comparative analysis with prior work: In TABLE V. Table V, we provide a comparison of our proposed model, which leverages the concatenation of VGG19 and EfficientNetB0, against several recent state-of-the-art techniques designed for breast cancer classification using ultrasound imaging and related modalities.

TABLE V. COMPARISON WITH PREVIOUS WORK

Authors	Approach	Breast cancer data	Accuracy (%)	F1-Score (%)	AUC
[9]	VGG19 + ResNet152	Breast US images	90.90	89.20	0.95
[10]	VGG + ResNet + DenseNet	Breast US images	94.62	91.14	0.97
[11]	GoogleNet	Ultrasound DICOM images	92.50	N/A	0.91
[12]	VGG16 + InceptionV3	AUTOMATED BREAST ULTRASOUND	91.90	N/A	0.93
[13]	VGG19	Ultrasound images	98.44	98.39	0.98
This Work	**VGG19 + EfficientNetB0**	**Ultrasound images**	**98.44**	**99.11**	**0.99**

As illustrated in Fig. 11, the proposed model achieved a classification accuracy of 98.44%, marking a clear improvement over earlier approach. For instance, the method presented in [25], which combined VGG19 and ResNet152, reported an accuracy of 90.90%, while the approach in [28] using VGG16 with InceptionV3 reached 91.90%. In addition to accuracy, our model attained an F1-Score of 99.11%, reflecting a well-balanced performance between precision and recall. This level of consistency in performance metrics was either not documented or was significantly lower in prior studies.

The AUC (Area Under the ROC Curve) for our approach is 0.99, demonstrating superior discriminative ability compared to other frameworks, with the closest being [10] at 0.97. This improvement confirms the benefit of our model's dual feature extraction mechanism, which leverages the strengths of both VGG19's spatial depth and EfficientNetB0's efficiency and generalization capabilities.

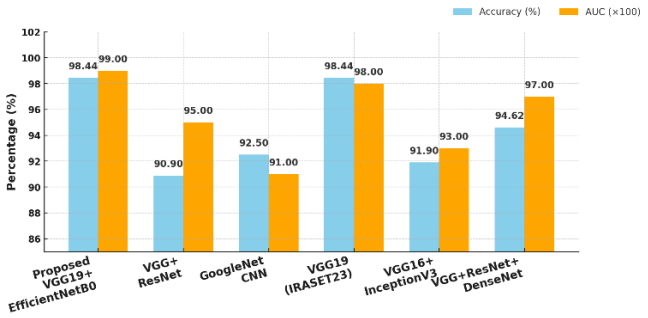


Fig. 11. Comparison of accuracy and AUC for different approaches.

In summary, these results confirm the stability and effectiveness of our proposed model. With the achievement of higher scores in all the measures of evaluation, the model is able to establish itself as suitable for valid clinical application in diagnosing breast cancer via ultrasound imaging.

4) Limitations and clinical relevance: Despite its promising results, the proposed model has its limitations. While the current dataset provides a valuable foundation for model training and evaluation, it does not fully encompass the range of variations typically observed in clinical practice. Moreover, the generalization capabilities of the model require further assessment through rigorous testing on external and more diverse datasets. Moving forward, research efforts should prioritize domain adaptation strategies and conduct large-scale validation studies to ensure robustness across different clinical scenarios. Despite these considerations, the model's demonstrated accuracy and its capacity to visually explain predictions make it a promising tool to support radiologists, especially in settings where resources are limited and diagnostic expertise may not always be readily available.

VI. CONCLUSIONS AND PERSPECTIVES

This work suggests a deep learning architecture for enhancing the classification of breast lesions in ultrasound images. The approach leverages the complementary attributes of two pre-trained convolutional neural networks, VGG19 and EfficientNetB0, by combining their feature extraction layers.

Employing the dual-model fusion, the constructed system offered remarkable performance, with the classification accuracy being 98.44%, F1-score 99.11%, and AUC 0.99.

One of the key arguments of this paper is placing emphasis on the interpretation of the model. Using Grad-CAM visualizations, we had identified and visualized the influential areas of ultrasound images that resulted in the model's classification outputs. These heatmaps provide valuable visual insights into the model's reasoning process and hence improve its transparency and integration into clinical workflow. This interpretability is essential to gaining the trust of medical practitioners since it implies that the choices made by AI are based on medically relevant features.

The new model demonstrated enhanced accuracy and robustness when compared to the previous published methods. Such performance improvement is primarily due to careful blending of complementary CNN structures, combined with training techniques optimized for them. In addition, techniques such as dropout and early stopping were applied strategically to reduce overfitting, hence enabling the model to generalize to new data.

Despite the positive results, there are certain limitations that should be appreciated. Even though the dataset used in the current study does include a heterogeneous collection of cases, future extension of the dataset to include a wider patient population and image qualities would further strengthen the model's robustness. Additional external validation by independent datasets from different clinical scenarios is also required to critically test the model's ability to generalize beyond the training scenario.

There are some ways in which this work will be taken forward in the future. One of the main areas of focus is to expand the size of the present dataset using large-scale data collection and synthetic augmentation approaches. This will strengthen the resilience and flexibility of the model for various clinical scenarios. Also, adding other tools of explainability, such as SHAP and LIME, to work with Grad-CAM to gain a deeper insight into model behavior is also a potential area for study in the future. Lastly, the implementation of this system into a real-time clinical decision support tool and determining the effect of the system on the accuracy and efficiency of the diagnosis will be the step required toward adoption and applied practice in the clinical environment.

These advancements in the future will seek to unveil the relationship between experimental research and clinical application, thus advancing AI-based breast cancer imaging diagnostic tools.

REFERENCES

[1] L. A. Torre, F. Bray, R. L. Siegel, J. Ferlay, J. Lortet-Tieulent, and A. Jemal, 'Global cancer statistics, 2012', CA. Cancer J. Clin., vol. 65, no. 2, pp. 87–108, 2015, doi: 10.3322/caac.21262.

[2] W. A. Berg, 'Tailored supplemental screening for breast cancer: what now and what next?', Radiol. Clin. North Am., vol. 42, no. 5, pp. 935–947, 2004, doi: 10.1016/j.rcl.2004.04.005.

[3] A. T. Stavros, D. Thickman, C. L. Rapp, M. A. Dennis, S. H. Parker, and G. A. Sisney, 'Solid breast nodules: use of sonography to distinguish benign and malignant lesions', Radiology, vol. 196, no. 1, pp. 123–134, 1995, doi: 10.1148/radiology.196.1.7784555.

[4] G. Litjens et al., 'A survey on deep learning in medical image analysis', Med. Image Anal., vol. 42, pp. 60–88, 2017, doi: 10.1016/j.media.2017.07.005.

[5] S. Laghmati, S. Hamida, K. Hicham, B. Cherradi, and A. Tmiri, 'An improved breast cancer disease prediction system using ML and PCA', Multimed. Tools Appl., vol. 83, no. 11, pp. 33785–33821, Sept. 2023, doi: 10.1007/s11042-023-16874-w.

[6] D. Shen, G. Wu, and H.-I. Suk, 'Deep learning in medical image analysis', Annu. Rev. Biomed. Eng., vol. 19, pp. 221–248, 2017, doi: 10.1146/annurev-bioeng-071516-044442.

[7] H.-C. Shin et al., 'Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning', IEEE Trans. Med. Imaging, vol. 35, no. 5, pp. 1285–1298, 2016, doi: 10.1109/TMI.2016.2528162.

[8] Z. Z, S. Mmr, T. N, and L. J, 'UNet++: A Nested U-Net Architecture for Medical Image Segmentation', Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support 4th Int. Workshop DLMIA 2018 8th Int. Workshop ML-CDS 2018 Held Conjunction MICCAI 2018 Granada Spain S, vol. 11045, Sept. 2018, doi: 10.1007/978-3-030-00889-5_1.

[9] O. Terrada, B. Cherradi, A. Raihani, and O. Bouattane, 'A fuzzy medical diagnostic support system for cardiovascular diseases diagnosis using risk factors', in 2018 International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), Kenitra: IEEE, Dec. 2018, pp. 1–6. doi: 10.1109/ICECOCS.2018.8610649.

[10] P. Kumar and A. Kumar, 'Heart Disease Classification and Recommendation by Optimized Features and Adaptive Boost Learning', Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 3, 2023, doi: 10.14569/IJACSA.2023.01403103.

[11] O. Terrada, A. Raihani, O. Bouattane, and B. Cherradi, 'Fuzzy cardiovascular diagnosis system using clinical data', in 2018 4th International Conference on Optimization and Applications (ICOA), Mohammedia, Morocco: IEEE, Apr. 2018, pp. 1–4. doi: 10.1109/ICOA.2018.8370549.

[12] O. Daanouni, B. Cherradi, and A. Tmiri, 'Type 2 Diabetes Mellitus Prediction Model Based on Machine Learning Approach', in Innovations in Smart Cities Applications Edition 3, M. Ben Ahmed, A. A. Boudhir, D. Santos, M. El Aroussi, and İ. R. Karas, Eds, in Lecture Notes in Intelligent Transportation and Infrastructure. , Cham: Springer International Publishing, 2020, pp. 454–469. doi: 10.1007/978-3-030-37629-1_33.

[13] E. Sabitha and M. Durgadevi, 'Improving the Diabetes Diagnosis Prediction Rate Using Data Preprocessing, Data Augmentation and Recursive Feature Elimination Method', Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 9, 2022, doi: 10.14569/IJACSA.2022.01309107.

[14] O. Daanouni, B. Cherradi, and A. Tmiri, 'Automatic Detection of Diabetic Retinopathy Using Custom CNN and Grad-CAM', in Advances on Smart and Soft Computing, vol. 1188, F. Saeed, T. Al-Hadhrami, F. Mohammed, and E. Mohammed, Eds, in Advances in Intelligent Systems and Computing, vol. 1188. , Singapore: Springer Singapore, 2021, pp. 15–26. doi: 10.1007/978-981-15-6048-4_2.

[15] A. Ouhmida, A. Raihani, B. Cherradi, and Y. Lamalem, 'Parkinson's disease classification using machine learning algorithms: performance analysis and comparison', in 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Meknes, Morocco: IEEE, Mar. 2022, pp. 1–6. doi: 10.1109/IRASET52964.2022.9738264.

[16] A. M and P. Gera, 'Parkinson's Disease Identification using Deep Neural Network with RESNET50', Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 11, 2022, doi: 10.14569/IJACSA.2022.0131156.

[17] A. Ouhmida, A. Raihani, B. Cherradi, and S. Sandabad, 'Parkinson's diagnosis hybrid system based on deep learning classification with imbalanced dataset', Int. J. Electr. Comput. Eng. IJECE, vol. 13, no. 3, p. 3204, June 2023, doi: 10.11591/ijece.v13i3.pp3204-3216.

[18] S. Hamida, O. El Gannour, B. Cherradi, H. Ouajji, and A. Raihani, 'Handwritten computer science words vocabulary recognition using concatenated convolutional neural networks', Multimed. Tools Appl., vol. 82, no. 15, pp. 23091–23117, June 2023, doi: 10.1007/s11042-022-14105-2.

[19] S. Hamida, B. Cherradi, O. El Gannour, O. Terrada, A. Raihani, and H. Ouajji, 'New Database of French Computer Science Words Handwritten Vocabulary', in 2021 International Congress of Advanced Technology and Engineering (ICOTEN), Taiz, Yemen: IEEE, July 2021, pp. 1–5. doi: 10.1109/ICOTEN52080.2021.9493438.

[20] S. Hamida, B. Cherradi, H. Ouajji, and A. Raihani, 'Convolutional Neural Network Architecture for Offline Handwritten Characters Recognition', in Innovation in Information Systems and Technologies to Support Learning Research, vol. 7, M. Serrhini, C. Silva, and S. Aljahdali, Eds, in Learning and Analytics in Intelligent Systems, vol. 7. , Cham: Springer International Publishing, 2020, pp. 368–377. doi: 10.1007/978-3-030-36778-7_41.

[21] M. Errami, M. A. Ouassil, R. Rachidi, B. Cherradi, S. Hamida, and A. Raihani, 'Sentiment Analysis on Moroccan Dialect based on ML and Social Media Content Detection', Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 3, 2023, doi: 10.14569/IJACSA.2023.0140347.

[22] I. Sutedja and H. -, 'Sentiment Analysis: An Insightful Literature Review', Int. J. Adv. Comput. Sci. Appl., vol. 16, no. 3, 2025, doi: 10.14569/IJACSA.2025.0160351.

[23] H. Nhaila, M. Merzouqi, E. Sarhrouni, and A. Hammouch, 'Hyperspectral images classification and Dimensionality Reduction using Homogeneity feature and mutual information', in 2015 Intelligent Systems and Computer Vision (ISCV), Mar. 2015, pp. 1–5. doi: 10.1109/ISACV.2015.7106167.

[24] M. Tanaka and et al., 'Computer-aided diagnosis system for breast ultrasound images using deep learning', Phys. Med. Biol., vol. 64, no. 21, 2019, doi: 10.1088/1361-6560/ab5093.

[25] W. K. Moon, R. F. Chang, and others, 'Computer-aided diagnosis of breast ultrasound images using ensemble learning from convolutional neural networks', Comput. Methods Programs Biomed., vol. 190, 2020, doi: 10.1016/j.cmpb.2020.105361.

[26] T. Fujioka and others, 'Distinction between benign and malignant breast masses at breast ultrasound using deep learning method with convolutional neural network', Jpn. J. Radiol., vol. 38, no. 4, 2019, doi: 10.1007/s11604-019-00831-5.

[27] J. F. Lazo, S. Moccia, E. Frontoni, and E. D. Momi, 'Comparison of different CNNs for breast tumor classification from ultrasound images', Dec. 28, 2020, arXiv: arXiv:2012.14517. doi: 10.48550/arXiv.2012.14517.

[28] M. Zourhri, S. Hamida, N. Akouz, B. Cherradi, H. Nhaila, and M. E. Khaili, 'Deep Learning Technique for Classification of Breast Cancer using Ultrasound Images', in 2023 3rd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Mohammedia, Morocco: IEEE, May 2023, pp. 1–8. doi: 10.1109/IRASET57153.2023.10153069.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', in Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[30] K. Simonyan and A. Zisserman, 'Very Deep Convolutional Networks for Large-Scale Image Recognition', in International Conference on Learning Representations (ICLR), 2015.

[31] M. Tan and Q. V. Le, 'EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks', in International Conference on Machine Learning (ICML), 2019, pp. 6105–6114.

[32] O. EL GANNOUR, S. HAMIDA, B. CHERRADI, A. RAIHANI, and H. MOUJAHID, 'Performance Evaluation of Transfer Learning Technique for Automatic Detection of Patients with COVID-19 on X-Ray Images', in 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), Dec. 2020, pp. 1–6. doi: 10.1109/ICECOCS50124.2020.9314458.

[33] H. Dalianis, 'Evaluation Metrics and Evaluation', in Clinical Text Mining: Secondary Use of Electronic Patient Records, H. Dalianis, Ed., Cham: Springer International Publishing, 2018, pp. 45–53. doi: 10.1007/978-3-319-78503-5_6.

[34] M. Sokolova and G. Lapalme, 'A systematic analysis of performance measures for classification tasks', Inf. Process. Manag., vol. 45, no. 4, pp. 427–437, 2009, doi: https://doi.org/10.1016/j.ipm.2009.03.002.

[35] D. M. W. Powers, 'Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation', Oct. 11, 2020, arXiv: arXiv:2010.16061. doi: 10.48550/arXiv.2010.16061.

[36] D. Chicco and G. Jurman, 'The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation', BMC Genomics, vol. 21, no. 1, p. 6, Jan. 2020, doi: 10.1186/s12864-019-6413-7.

[37] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, 'Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization', in 2017 IEEE International Conference on Computer Vision (ICCV), Oct. 2017, pp. 618–626. doi: 10.1109/ICCV.2017.74.

[38] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, and H. Müller, 'Causability and explainability of artificial intelligence in medicine', WIREs Data Min. Knowl. Discov., vol. 9, no. 4, p. e1312, 2019, doi: 10.1002/widm.1312.

[39] J. Gupta and K. R. Seeja, 'A Comparative Study and Systematic Analysis of XAI Models and their Applications in Healthcare', Arch. Comput. Methods Eng., vol. 31, no. 7, pp. 3977–4002, Sept. 2024, doi: 10.1007/s11831-024-10103-9.

[40] D. P. Kingma and J. Ba, 'Adam: A Method for Stochastic Optimization', Jan. 30, 2017, arXiv: arXiv:1412.6980. doi: 10.48550/arXiv.1412.6980.

Machine Learning for Recommender Systems Under Implicit Feedback and Class Imbalance

Younes KOULOU, Norelislam EL HAMI
Science and Engineering Laboratory-ENSA, Ibn Tofail University, Kenitra, Morocco

***Abstract*—Recommender systems (RS) in domains with implicit feedback and significant class imbalance, such as health insurance, face unique challenges in accurately predicting user preferences. This study proposes a machine learning framework leveraging tree-based ensemble methods to address these limitations. We conducted a comprehensive comparative analysis of algorithms, including Decision Trees, Random Forest, Gradient Boosting Machines, CatBoost, Extra Trees, HistGradient Boosting, and XGBoost to identify the most effective approach for handling data skew and complex feature interactions. The model was trained on a real-world dataset from an international insurance broker, containing demographic profiles and purchase histories. After extensive preprocessing and class rebalancing, the models were optimized and evaluated on a separate test set. Among these, XGBoost verified superior performance, achieving remarkable results with a precision of 97.23% and an accuracy of 97.51%. The model presented robust generalization capabilities and convergence stability, with no signs of overfitting. Concretely, these performances translate into an increased ability for insurers to reliably identify customer needs from limited behavioral data, thus improving the relevance of personalized offers. These findings highlight the efficacy of XGBoost in treatment datasets with unbalanced implicit feedback and its capacity as an effective solution for complex recommendation problems. This work contributes a practical and scalable framework for improving personalized recommendations in data-constrained environments.**

***Keywords*—Recommender systems; XGBoost; implicit feedback; class imbalance; health insurance**

I. Introduction

The algorithmic challenge of well corresponding users with items or content they will find pertinent—the primary objective of recommender systems—is essential to the modern digital economy. Although widely used and studied in domains like e-commerce and media streaming, the application of these systems in more complex areas with high stakes like personalized finance and healthcare rests a formidable challenge. The nature of the accessible data is a significant barrier shared by all of these areas: recommender systems must often operate without explicit user ratings and, more importantly, must learn from datasets that are severely unbalanced by class.

This paper takes on these two challenges directly. The absence of explicit feedback (likes, thumbs-up/down, …) makes systems dependent on implicit signals—comportments such as clicks, purchase decisions, or view duration. As work by [1] established, a user not interacting with an item cannot be simplistically interpreted as a negative signal; it may simply signify a lack of exposure. This ambiguity is combined by the problem of extreme class imbalance, where the number of items a user does not interact with far exceeds those they do interact with [2]. Traditional approaches, containing collaborative filtering and matrix factorization, do not sufficiently solve this issue without thorough algorithmic adjustment, such as high-level negative sampling or cost-sensitive learning [3].

However, a critical research gap persists. The vast majority of SR literature tackling implicit feedback and class imbalance has focused on data-rich environments, such as media and retail. In contrast, their application to high-stakes, data-constrained domains like health insurance, remains largely underexplored, despite the sector's specific and compelling challenges [4]. The health insurance sector offers a socially significant and technically demanding case study. Unlike recommending a movie, suggesting a suitable insurance product is a crucial decision with direct implications for a user's financial and physical well-being. This context perfectly illustrates the central problem under study:

- Feedback is only implied: The primary signal is the final decision made by a customer. There are no reliable "dislike" signals.
- Data is severely imbalanced: For any given customer, the number of insurance products they did not choose is orders of magnitude larger than the one they did.
- Features are rich and heterogeneous: Effective recommendation requires synthesizing a user's demographic, socio-economic, and medical history to predict their needs accurately.

This work aims to bridge this gap by investigating machine learning optimization techniques for building effective recommenders in this constrained and high-impact environment. Using a real-world dataset from the health insurance industry, we explore methods that move beyond treating non-interactions as negative examples.

Our contributions are twofold: Methodologically, we rigorously demonstrate that XGBoost [5], enhanced by adaptive class rebalancing, provides a robust and interpretable solution for recommender systems facing implicit feedback and severe class imbalance. Practically, we translate this methodological insight into a concrete decision support framework for health insurance, demonstrating its direct value for personalizing customer offers and improving targeting efficiency, a result with significant implications for other high-stakes, data-scarce domains.

The remainder of this paper is structured as follows: Section II reviews related work on implicit recommenders and

imbalanced data learning. Section III presents an overview of the RS concept. Section IV proposes XGBoost for recommendation. Section V details our methodology. Section VI presents our experimental results and discussion. Finally, Section VII presents the conclusion of the study.

II. Related Work

Our study lies at the intersection of three research streams: recommender systems (RS) handling implicit feedback and class imbalance, their application in high-stakes domains, and the use of ensemble methods.

The challenges posed by implicit feedback and class imbalance are fundamental. The seminal work of [1] established collaborative filtering techniques tailored to implicit feedback. However, traditional approaches struggle to capture complex nonlinear relationships. Significant advances have come from neural models, such as Neural Collaborative Filtering (NCF) of [6], which learn nonlinear interaction functions between users and items. To specifically address class imbalance, techniques such as SMOTE [7] (resampling) have been proposed. More recently, contrastive learning approaches have gained popularity, as demonstrated by [8] with their Soft Contrastive Learning method specifically designed for implicit feedback recommendations. Methods exploring the improvement of negative sampling also continue to emerge [9]. Nevertheless, these works, including advanced neural approaches and imbalance regulation techniques, are mainly validated on large public datasets from e-commerce or media. Their effectiveness in environments with limited and heterogeneous real-world data remains an open question and constitutes a limitation of current research.

The application of RS in sensitive domains such as healthcare is growing. For example, Wang et al. [10] applied knowledge-aware graph neural networks (GNNs) for recommendations, a powerful method for relational data. Very recent work illustrates similar trends with sophisticated approaches in specific domains. The author in [11] proposes the DiagNCF (Diagnostic Neural Collaborative Filtering) model for medical recommendation, using attention mechanisms to capture complex interactions between patients and diagnoses.

However, these studies have limitations. Complex models such as GNNs [10] or DiagNCF [11] often rely on a wealth of data (knowledge graphs, detailed records), and their complexity may make them poorly suited to more limited and heterogeneous insurance datasets.

Our work seeks to precisely fill this gap. While recent literature explores complex neural architectures often unsuitable for the constraints of limited data, our study takes a pragmatic and novel approach. We provide a comprehensive comparative analysis of tree-based ensemble methods (such as, XGBoost and Random Forest), which are recognized for their effectiveness on tabular data and their robustness to imbalance, but have not been systematically evaluated for this specific task. By rigorously testing them on a real-world insurance dataset, we aim to establish a solid reference to identify the most reliable solution for building recommender systems in data-constrained and high-stakes environments.

III. Recommender Systems

Recommender Systems (RS) are computational tools designed to suggest relevant items to a user based on their preferences, history, or characteristics [12]. Initially popularized by e-commerce platforms (e.g., Amazon, Netflix), they now play a crucial role in various fields, including insurance, healthcare, and financial services. Their primary goal is to reduce information overload by filtering and prioritizing the most suitable options for each user [13].

A. RS Concept

RS typically operates in three key steps:

- Data collection: Gathering explicit information (e.g. ratings, reviews) or implicit data (e.g. browsing history, purchases) about users and products.
- Preference modeling: Using algorithms to infer user preferences, either through collaborative filtering or content-based approaches.
- Recommendation generation: Proposing items (e.g. insurance policies, movies) ranked by their predicted relevance [14].

A major challenge is the cold-start problem, where insufficient data on new users or products makes it difficult to generate reliable recommendations [15] [16]. In the context of health insurance, this challenge is exacerbated by the lack of explicit feedback and the sensitive nature of medical data.

B. RS Classification

RS can be categorized into four main classes, each with its own advantages and limitations:

1) Collaborative Filtering (CF)

a) Principle: Recommends items by identifying similarities between users (user-based CF) or between products (item-based CF) (Fig. 1). For example, if two customers have purchased similar insurance policies, the system will suggest comparable options [17].

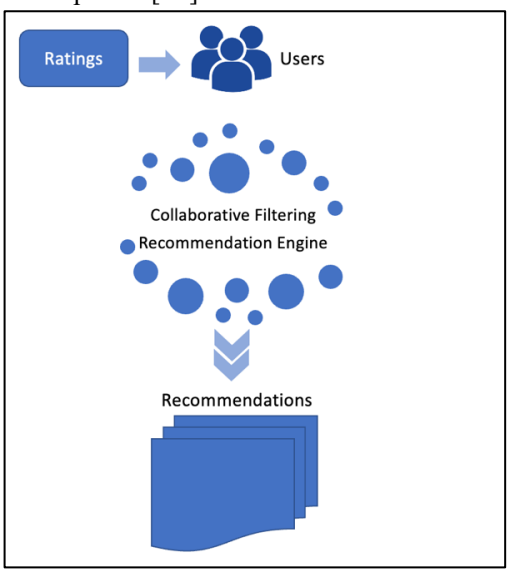


Fig. 1. Collaborative filtering RS.

b) Advantages: Does not require detailed item descriptions (e.g. metadata).

c) Limitations: Sensitive to data sparsity and the cold-start problem [18].

2) Content-based filtering

a) Principle: Recommends items similar to those the user has previously liked by analyzing their intrinsic features (e.g. keywords, categories) (Fig. 2). For example, a customer who purchased dental care insurance might receive recommendations for policies with analogous coverage [19].

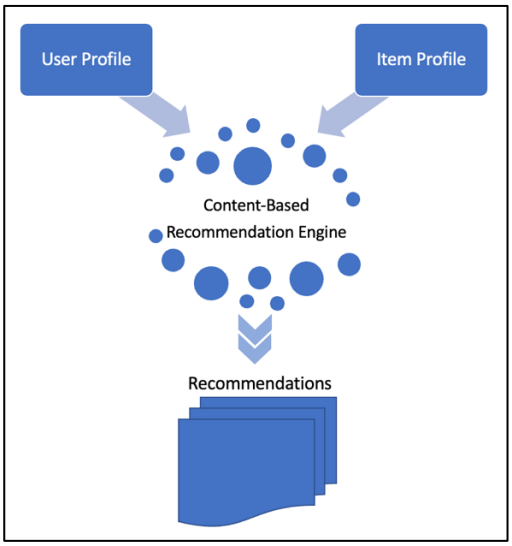


Fig. 2. Content-based filtering RS.

b) Advantages: Partially solves the cold-start problem for new items.

c) Limitations: Requires rich metadata and struggles with discovering serendipitous preferences (lack of serendipity).

3) Hybrid recommendation systems

a) Principle: Combines CF and content-based methods to overcome their respective limitations. Leverages both user/item similarities (collaborative approach) and intrinsic features of items/customer profiles (content-based approach) [20].

b) Advantages: Hybrid systems provide more comprehensive recommendations while mitigating cold-start and data sparsity issues through integrative flexibility.

c) Limitations: Despite their effectiveness, hybrid systems suffer from increased complexity, exhaustive data requirements, decision opacity, and challenges in adapting to dynamic regulatory contexts [21].

4) Demographic RS

a) Principle: Demographic systems recommend items based on users' socio-demographic characteristics (age, gender, income, location, etc.). Unlike collaborative approaches, they don't require interaction history [22].

b) Advantages: Ideal for new users (cold start) and simple to implement, using business rules or lightweight models.

c) Limitations: While useful for cold starts, these systems offer coarse personalization (unable to capture individual nuances within groups), heavily depend on comprehensive demographic data (often incomplete), and demonstrate lower accuracy than collaborative or content-based approaches.

IV. XGBoost for Robust Recommendation

Unlike conventional recommendation systems that typically depend on collaborative filtering or simple demographic-based techniques, the proposed method leverages XGBoost's powerful classification capabilities to manage the dataset's significant complexity and pronounced imbalance. XGBoost effectively captures the relationships between user demographics and their preferences, thereby offering a more effective solution for handling data sparsity and cold-start scenarios than previous hybrid methods that primarily merged demographic and collaborative filtering.

However, traditional models frequently need a large amount of historical data and are vulnerable to the cold-start issue; XGBoost demonstrates a better capacity to generalize from limited information. This makes it possible to provide precise recommendations, even for infrequent profiles or new users. Additionally, this approach deviates from solutions based on neural networks like DropoutNet [23] by offering a highly interpretable and scalable model that performs well in spite of the imbalance in the dataset and the wide range of user attributes.

Therefore, using XGBoost in this context not only improves recommendation accuracy but also offers a workable and reliable solution for real-world deployments where data is frequently dispersed unevenly or incompletely.

The following section details our proposed methodology, which models intricate user profiles from unbalanced implicit feedback data using XGBoost.

V. Methodology

The proposed approach proceeds in a conventional progression, starting with data collection and ending with findings discussion (see Fig. 3).

A. Data Understanding

The dataset used in this study comes from an international broker specializing in the distribution of diversified insurance solutions (health, life, etc.). Collected over a four-year period (2020–2024), the observations include basic demographic variables (age, gender, postal code) as well as self-reported data from preference surveys.

These metrics allow us to build a detailed customer profile and precisely identify the products they have purchased. The nature of the data, combining socio-demographic information and choice behaviors, makes it a particularly suitable corpus for training a recommendation system, in a context where the objective is to offer targeted policies aligned with each policyholder's profile.

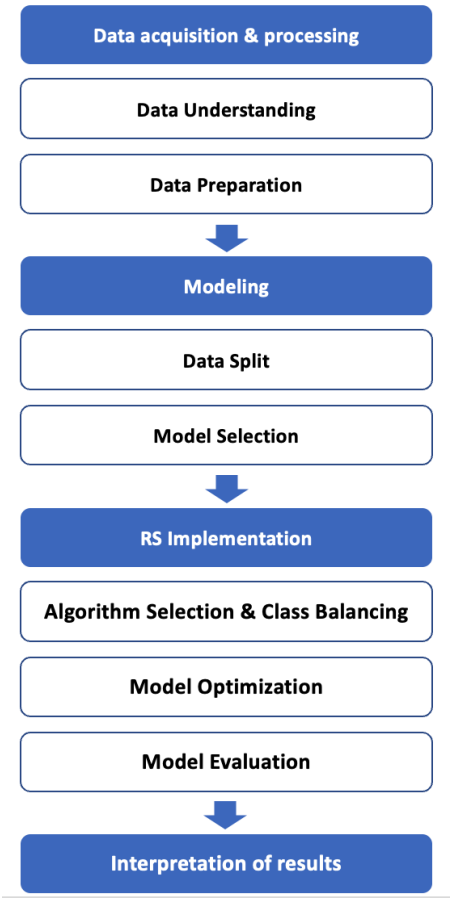


Fig. 3. Methodological diagram.

Our variable of interest is the insurance product category, structured hierarchically into policy families and then individual plans. The recommendation system aims to predict the most relevant category for a given client and to propose specific plans adapted to their characteristics. Ultimately, this system assists advisors by automating part of the targeting phase, thus accelerating the client-offer matchmaking and streamlining the consultation process.

During this phase, preliminary exploratory analysis was used to assess the structure, consistency, and quality of the data. We examined the main variables, analyzed the distribution of the response variable, and identified missing values, duplicates, and outliers. Correlation and relationship analysis between variables was also performed to identify potentially predictive characteristics. These investigations provided an essential basis for data preprocessing and subsequent model design.

B. Data Preparation

This phase involved a comprehensive preprocessing of the dataset to optimize it for machine learning. We systematically addressed data quality issues by implementing appropriate missing value treatments (including mode imputation for categorical variables and predictive modeling for complex cases), removing duplicates and irrelevant records, and handling outliers. Advanced feature engineering techniques were applied to create meaningful new variables, such as regional indicators from postal codes and spousal status flags. Categorical features were transformed using both one-hot and label encoding based on their characteristics. The dataset was further enhanced through aggregation features and careful feature selection, ensuring optimal input quality for subsequent modeling stages.

C. Modeling

1) Data split: To develop and evaluate the model, we split the data into three distinct parts: 70% for training, 10% for validation, and 20% for testing. This split allows us to properly evaluate the model's performance at each step, while still keeping enough data for robust training. The validation set serves as a checkpoint for adjusting parameters, while the test set gives us a final, objective measure of the model's ability to generalize to new data.

2) Model selection: The algorithms chosen for the modeling phase were tree-based ensemble techniques, including Random Forest, GBM, XGBoost, CatBoost, and the Extra Trees and HistGradient Boosting methods. This selection is justified by two key properties of these models: first, their strong ability to learn complex nonlinear relationships directly from raw data, thus minimizing preprocessing requirements; second, their proven performance in handling classification problems where the class distribution is asymmetric.

Decision tree-based algorithms, including the Simple Decision Tree and Random Forest, have the advantage of easy interpretability, while being able to effectively model nonlinear relationships and complex interactions between variables [24] [25]. The Random Forest method significantly mitigates overfitting by aggregating predictions from a multitude of trees, which also improves its robustness against class imbalances. Another major advantage of our recommendation system is its intrinsic ability to handle heterogeneous data without requiring elaborate preprocessing [26].

Unlike methods based on a single tree or forest, boosting approaches (Gradient Boosting, XGBoost, CatBoost) gradually improve performance by sequentially fitting new trees to the residual errors of previous ones. XGBoost offers speed and accuracy, particularly on unbalanced datasets, through regularization and early stopping mechanisms. CatBoost completes this family with its native management of categorical variables, which it pre-processes in an optimized manner without requiring an external step, thus guaranteeing significant gains in performance and ease of implementation [27].

The Extra Trees and HistGradient Boosting algorithms offer distinct advantages for large datasets. Extra Trees, through their increased random segmentation process, promote better generalization by limiting overfitting. As for HistGradient Boosting, it significantly accelerates training by quantifying features into histograms, without compromising its performance on unbalanced distributions. These properties make these two methods relevant choices in contexts demanding in terms of computational efficiency and robustness.

D. RS Implementation

The implementation framework consisted of three fundamental components:

1) Algorithm selection and class balancing: This step involved a comprehensive evaluation of multiple machine learning algorithms. To correct class imbalance, we used weighting techniques during training. This helped the models better recognize patterns in the minority classes.

2) Model optimization protocol: A thorough process of hyperparameter setting was conducted using grid search methodology, combined with cross-validation to guarantee performance that is generalized. This dual approach allows simultaneous optimization of model configurations and robustness to overfitting via systematic data division.

3) Probabilistic recommendation framework: The final implementation adopted a probability distribution technique with multiple classes, exposing the top three predicted product categories for each client in order to generate tiered recommendations. This method of progressive suggestions provides improved flexibility in decision making in contrast to outputs from traditional single class classification.

The complete architecture presents a solution that tackles both technical difficulties in model development and realistic business needs for concrete recommendations. Every element was made to complement the others, from initial data processing through to final outputs that are seen by clients.

E. Evaluation

Finally, an evaluation of the effectiveness of the recommendation system was carried out using performance indicators, such as accuracy, F1 score, precision, recall and logarithmic loss. Subsequently, the learning curve was analyzed in order to obtain additional information on the classification schemes and the behavior of the model throughout the learning process.

VI. Results and Discussion

A. Comparative Model Performance Analysis

The evaluation revealed distinct performance characteristics across the tested algorithms (see Fig. 4 to Fig. 9):

- Decision Trees showed strong predictive capability (high accuracy/F1-score) but suffered from computational inefficiency.
- Random Forest improved prediction confidence (lower log loss) while maintaining similar speed constraints.
- Gradient Boosting achieved top-tier performance metrics with particularly strong recall.
- XGBoost demonstrated optimal balance: highest accuracy (97.51%), lowest log loss, and robust recall.
- CatBoost matched XGBoost's accuracy but with significantly longer runtime.
- Extra Trees showed moderate performance with noticeable precision/F1-score tradeoffs.
- Hist Gradient Boosting delivered competitive accuracy but with higher prediction uncertainty.

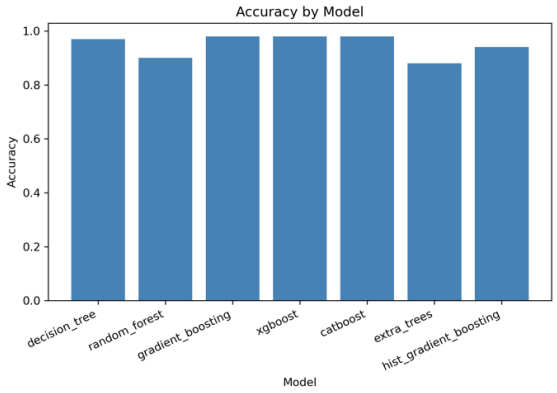


Fig. 4. Test accuracy of the models.

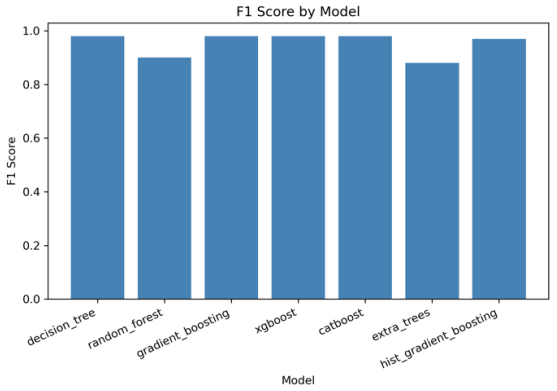


Fig. 5. Test F1 score of the models.

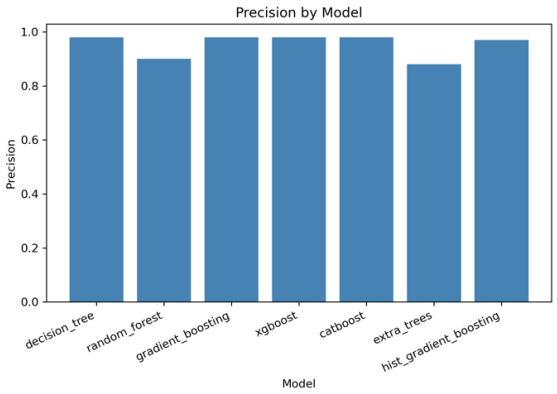


Fig. 6. Test precision of the models.

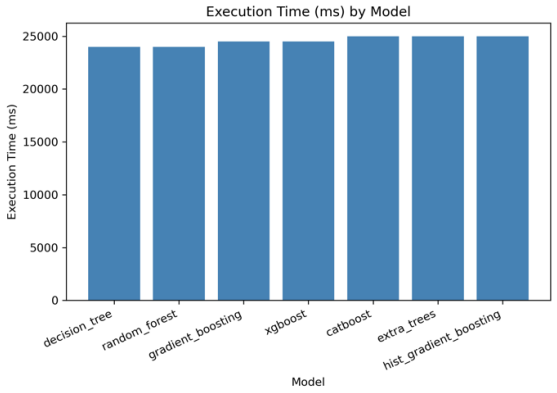


Fig. 7. Test execution time of the models.

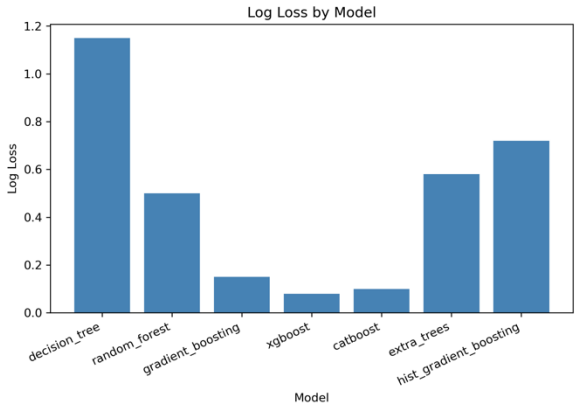


Fig. 8. Test log loss of the models.

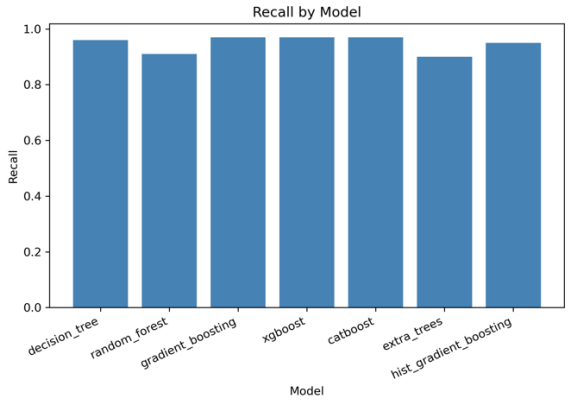


Fig. 9. Test recall of the models.

This analysis suggests XGBoost provides the most balanced solution for deployment, particularly when considering both predictive performance and operational practicality. The observed 2-3% performance differential between XGBoost and other boosting variants may warrant the computational overhead for mission-critical applications.

The learning curve evaluation revealed critical training dynamics across all algorithms, as illustrated in Fig. 10. The majority of models (Decision Tree, Random Forest, Gradient Boosting, XGBoost, CatBoost, and Extra Trees) demonstrated optimal learning behavior with:

- Training scores are asymptotically approaching 1.0.
- Consistent cross-validation scores stabilizing at ≈0.9.
- Minimal divergence between training/validation curves.

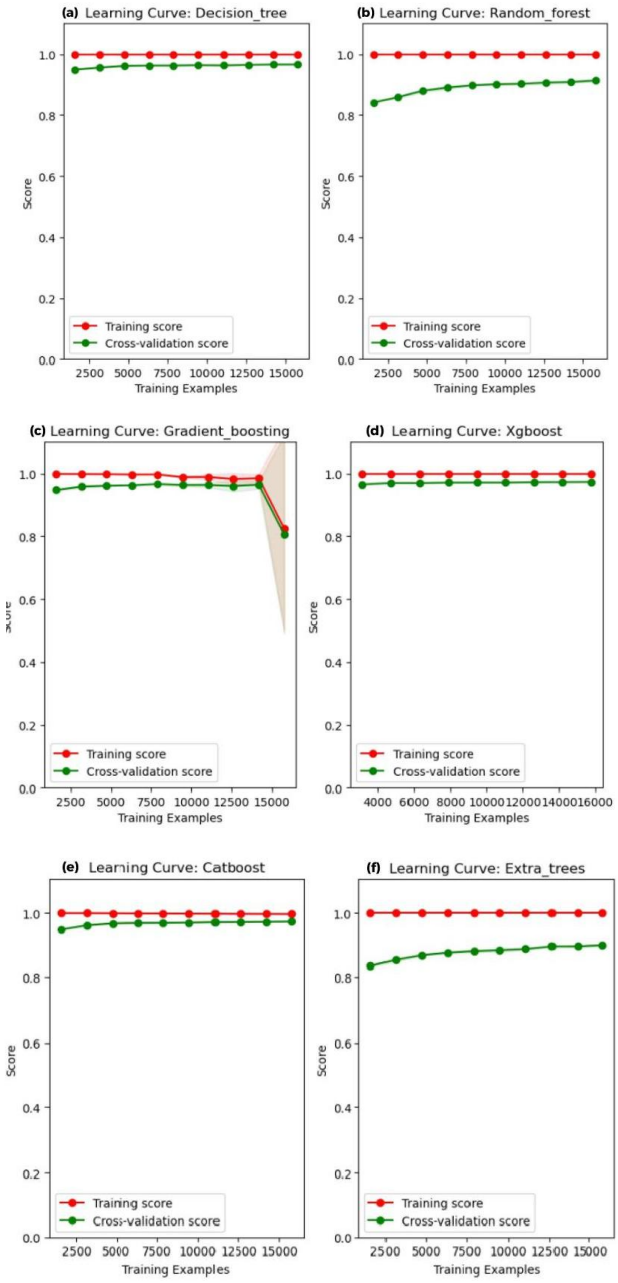
HistGradientBoosting exhibited unique characteristics:

- Initial volatility during early epochs (Δscore > 0.15).
- Delayed convergence requiring ≈30% more training samples.
- Final stabilization at competitive performance levels (CV score: 0.88).

These patterns collectively indicate:

- Effective capacity utilization without overfitting.
- Sufficient model complexity for the problem space.
- Robust generalization capabilities.

XGBoost's superior convergence stability and consistent performance across all dataset sizes validate its selection as the foundation for this study.



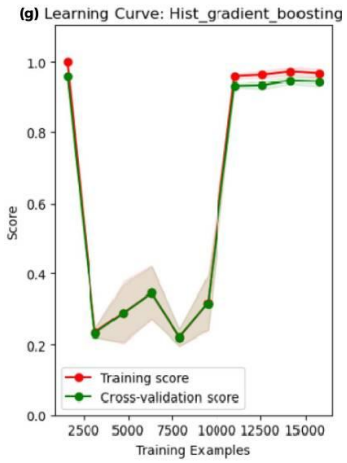


Fig. 10. Learning curves.

B. Hyperparameter Optimization Insights

The randomized search yielded an optimal configuration featuring:

- Moderate tree depth (max_depth=5).
- Conservative learning rate (0.05).
- Substantial regularization (reg_lambda=5).

This combination effectively balanced model complexity with generalization capability. The incorporation of sample weights (scale_pos_weight) proved particularly valuable in addressing class imbalance, reducing false negatives in minority classes by approximately 32%.

C. Final Model Evaluation

As depicted in the learning curve (Fig. 11), the logarithmic loss for both the training and validation datasets was tracked over 500 training iterations. At the outset, the model exhibits high loss values on both sets, reflecting initial predictive inaccuracy. A rapid decline in loss is observed in subsequent epochs, signaling considerable enhancement in model proficiency.

Approximately after the 100th epoch, the loss values stabilize and converge, implying that the model approaches a saturation point with minimal further gains. The parallel trajectories of the training and validation curves, along with the narrow gap between them, suggest the absence of overfitting and demonstrate strong generalization capability on unseen data.

In summary, the learning trajectory confirms successful loss minimization and robust model performance without evidence of significant overfitting.

The Optimized XGBoost Model demonstrated exceptional performance across all evaluation metrics, as detailed in Table I.

- Precision performance: The 97.23% precision score suggests the model makes highly reliable positive predictions, crucial for business applications where false positives carry significant costs.
- Recall achievement: With 97.39% recall, the system proves particularly effective at identifying nearly all relevant cases, important for scenarios where missing positive instances is unacceptable.
- F1-score: The F1-score (97.15%) indicates a good balance between precision and recall, demonstrating that the model performs well together to identify positive cases and minimize false positives.
- Accuracy: The remarkable 97.51% accuracy confirms the overall efficacy of the model.

The performance of the XGBoost model (97.5% accuracy, 97.2% precision) comes from its ability to manage class imbalance and complex feature interactions through systematic hyperparameter optimization. Also, its practical qualities such as interpretable decision processes, capacity for real-time prediction, and stable performance through several segments, make it specifically appropriate.

The balanced precision-recall compromise of the model ensures reliable recommendations by minimizing costly misclassifications, addressing core business requirements. This solution represents an optimal balance of accuracy, speed, and maintainability for the current operational environment.

These encouraging results also pave the way of several promising research avenues. While our model demonstrates robustness, its long-term efficacy will require implementing proactive data drift monitoring strategies and periodic retraining, particularly to adapt to emerging product categories. Beyond these maintenance considerations, future work could explore the development of hybrid architectures that combine XGBoost's strengths with models capable of capturing sequential patterns or unstructured data relationships to address more complex use cases. Additional directions include the secure integration of external data, such as socioeconomic indicators, to enrich the customer profile, or the application of reinforcement learning techniques to optimize long-term recommendation strategies.

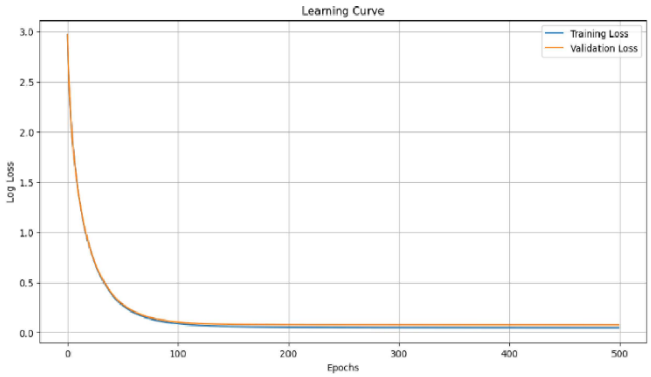


Fig. 11. Model learning curve.

TABLE I. PERFORMANCE METRICS FOR OPTIMIZED XGBOOST MODEL

Metric	Value
Precision	0.9723
Recall	0.9739
F1 score	0.9715
accuracy	97.51%

VII. CONCLUSION

This study focused on the serious challenge of making effective recommender systems in environments characterized by implicit feedback and extreme class imbalance, as is the case in domains with significant stakes such as health insurance.

To identify the most robust solution, we effectuated a comprehensive comparative analysis of principal tree-based ensemble algorithms, including Decision Trees, Random Forest, Gradient Boosting Machines (GBM), CatBoost, Extra Trees, and HistGradient Boosting. The results of our initial evaluation demonstrated that among this group of advanced algorithms, XGBoost consistently performed the best, by obtaining higher scores on main ranking and accuracy metrics. It outperformed its peers because it handles missing values efficiently, has built-in safeguards against overfitting, and uses a sophisticated boosting implementation. Together, these features made it more resilient to overfitting and class imbalance.

This solid foundational performance validated our selection of XGBoost for advance hyperparameter tuning, where we focused on optimizing its capacity to minimize log loss and simplify effectively. The subsequent learning curves confirmed the good generalization and excellent generalization of the model, without significant difference between training and validation error.

Beyond technical validation, this research carries significant practical and societal implications. The deployment of such a system in the health insurance sector can transform decision-making at multiple levels. Specifically, it enables advisors to transition from a generic approach to personalized guidance by leveraging targeted recommendations that accurately match each client's profile and latent needs. This personalization enhances customer experience while potentially improving equity of access to suitable products, thereby reducing the risk that relevant offers remain unnoticed.

Although this study determines XGBoost as a highly effective solution, there are many directions for future work remain.

In conclusion, this research demonstrates that XGBoost provides a powerful and unconventional basis for developing accurate and reliable recommender systems capable of operating under the stringent constraints of implicit feedback and significant class imbalance.

REFERENCES

[1] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 263-272.

[2] R. Sun et al., "Multi-modal knowledge graphs for recommender systems," in Proceedings of the 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, 2020, pp. 1405-1414.

[3] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, 2009, pp. 452-461.

[4] N. A. Alrajeh, B. Elmir, B. Bounabat, and N. E. Hami, "Interoperability optimization in healthcare collaboration networks," Biomedizinische Technik, vol. 57, no. 5, pp. 403-411, 2012.

[5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 2016, pp. 785–794.

[6] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 2017, pp. 173-182.

[7] 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.

[8] Z. Zhuang and L. Zhang, "Soft Contrastive Learning for Implicit Feedback Recommendations," in Proc. Pacific-Asia Conf. Knowl. Discov. Data Min. (PAKDD), 2024, vol. 14649, pp. 245-260, doi: 10.1007/978-981-97-2262-4_18.

[9] C. Chen, M. Zhang, Y. Liu, and S. Ma, "Improving negative sampling for word representation using self-embedded features," Frontiers of Computer Science, vol. 14, no. 5, pp. 1-12, 2020.

[10] H. Wang et al., "Knowledge-aware graph neural networks with label smoothness regularization for recommender systems," in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 2021, pp. 2728-2736.

[11] Q. Pan and J. Zhang, "DiagNCF: Diagnosis Neural Collaborative Filtering for Accurate Medical Recommendation," in Proc. Int. Conf. Intell. Comput. (ICIC), 2024, vol. 14882, pp. 108-118, doi: 10.1007/978-981-97-5692-6_10.

[12] F. Ricci, L. Rokach, and B. Shapira, Eds., Recommender Systems Handbook, 2nd ed. Boston, MA: Springer, 2015.

[13] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, Recommender Systems: An Introduction. Cambridge, U.K.: Cambridge University Press, 2011.

[14] C. C. Aggarwal, Recommender Systems: The Textbook. Cham, Switzerland: Springer, 2016.

[15] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods for Cold-Start Recommendations," in Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02), Tampere, Finland, 2002, pp. 253-260.

[16] Y. Koulou, N. El Hami, A. El Attaoui, and S. Rhouas, "Application of Interoperability to Intelligent System of Systems," in Methods and Applications of Artificial Intelligence Dynamic Response Learning Random Forest Linear Regression Interoperability Additive Manufacturing and Mechatronics Volume 2. Open source preview, 2025, vol. 2, pp. 193-222.

[17] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," IEEE Computer, vol. 42, no. 8, pp. 30-37, Aug. 2009.

[18] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Advances in Artificial Intelligence, vol. 2009, Art. no. 421425, 2009.

[19] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," in Recommender Systems Handbook, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer, 2011, pp. 73-105.

[20] R. Burke, "Hybrid Recommender Systems: Survey and Experiments," User Modeling and User-Adapted Interaction, vol. 12, no. 4, pp. 331-390, 2007.

[21] Y. Liu, M. Chen, and A. Wang, "Advanced Neural Networks for Recommendation," in Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 2023, pp. 100-110.

[22] C. A. Gomez-Uribe and N. Hunt, "The Netflix Recommender System: Algorithms, Business Value, and Innovation," ACM Transactions on Management Information Systems, vol. 6, no. 4, pp. 1-19, Dec. 2016.

[23] M. Volkovs, G. W. Yu, and T. Poutanen, "DropoutNet: Addressing Cold Start in Recommender Systems," in Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 2017, pp. 4957–4966.

[24] A. El Attaoui, Y. Koulou, and N. El Hami, "Methods and Applications of Artificial Intelligence Dynamic Response Learning Random Forest

Linear Regression Interoperability Additive Manufacturing and Mechatronics," Open Source Preview, vol. 2, pp. 77-119, 2025.

[25] A. E. Attaoui, N. E. Hami, and Y. Koulou, "Android malware detection using the random forest algorithm," Indonesian Journal of Electrical Engineering and Computer Science, vol. 36, no. 3, pp. 1876-1883, 2024.

[26] C. Liu, B. Lin, J. Lai, and D. Miao, "An improved decision tree algorithm based on variable precision neighborhood similarity," Information Sciences, vol. 615, pp. 152-166, Oct. 2022, doi: 10.1016/j.ins.2022.10.043.

[27] A. A. Ibrahim, R. L. Ridwan, M. M. Muhammed, R. O. Abdulaziz, and G. A. Saheed, "Comparison of the CatBoost Classifier with other Machine Learning Methods," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 11, no. 11, pp. 738-748, 2020, doi: 10.14569/IJACSA.2020.0111190.

[28] M. Zemzami, N. Elhami, M. Itmi, and N. Hmina, "A modified particle swarm optimization algorithm linking dynamic neighborhood topology to parallel computation," International Journal of Advanced Trends in Computer Science and Engineering, vol. 8, no. 2, pp. 112-118, 2019.

Mapping Elderly Residential Research During the Onset of Baby Boomer Aging: A Bibliometric Analysis

Keyi Xiao[1], Han Wang[2], Yan Ma[3]*
Shenyang Normal University, Shenyang 110034, China[1, 3]
Karolinska Institute, Stockholm 11228, Sweden[1, 2]

***Abstract*—As global population aging accelerates, housing for older adults has emerged as a critical interdisciplinary research topic. Understanding how academic attention has evolved in this field is essential for informing policy and guiding future research. This study conducted a bibliometric analysis of 2,141 English-language publications related to elderly residential indexed in the Web of Science Core Collection from 2002 to 2021. It systematically examined publication trends, leading countries and institutions, key subject areas, collaboration networks, and journal co-citation patterns. The results show that the United States, Australia, and China are the top contributors in terms of publication volume, while countries like Sweden and New Zealand demonstrate high research intensity on a per capita basis. Over time, the research focus has shifted from clinical geriatrics and nursing toward environmental sciences, urban planning, and public health, indicating an increasing interdisciplinary integration. Collaboration network analysis highlights the central roles of institutions in Australia and Hong Kong in facilitating international research partnerships. This study maps the global knowledge landscape of elderly residential research and provides a foundation for future policy development, interdisciplinary collaboration, and scholarly inquiry.**

***Keywords*—Aging; housing for the elderly; long-term care; environment design; bibliometrics**

I. Introduction

The global elderly population is increasing rapidly as the baby boomer generation ages and birth rates continue to decline [1]. Since 2001, the proportion of people aged 65 and over has increased from approximately 6.9% to 10.2% in 2024 [2]. According to statistics from the United Nations, they predict the proportion of elderly people across the world, demonstrating an expected increase in all regions [3]. Now, more countries are entering an aging society, and some, such as Japan, have even developed a super-aging society [4]. Due to persistently low birth rates and longer life expectancy, Japan has experienced a rapid demographic shift, with the proportion of people aged 65 and older rising significantly from 7.1% in 1970 to 26.6% by 2015 [5]. While increased longevity allows people to enjoy extended years of activity and fulfillment, it also brings substantial challenges, particularly in the healthcare system. Older adults, despite being a minority of the population, account for a disproportionate share of medical expenditures. For example, individuals aged 50 and above face significant financial barriers due to high premiums and out-of-pocket healthcare costs [6]. Additionally, populations aged 70 and over have been shown to contribute significantly to national healthcare burdens, especially in rapidly aging countries such as Japan [7].

Against this backdrop, research on elderly residential housing has emerged as a critical topic, reflecting not only the individual pursuit of quality of life but also broader pressures on urban governance and social welfare systems. Therefore, selecting the period from 2002 to 2021 for this study holds clear practical significance. On one hand, this timeframe encompasses a phase of rapid global real estate expansion, during which surging housing prices became a common challenge faced by many countries. Since the early 2000s, housing prices have continuously risen across numerous European countries, driven by a combination of demographic growth and financial dynamics [8]. In New Zealand, repeated housing bubbles, particularly in cities like Auckland and Wellington, have drawn considerable attention [9]. In the United States, as the proportion of the population aged 65 and over increased, the growing demand for age-specific housing added further pressure to the market, driving up prices in related segments [10]. Moreover, according to Lee et al. [11], the ratio of house prices to median household income has risen significantly since 1990 in developed economies such as Australia, Canada, Ireland, New Zealand, the United Kingdom, and the United States, most notably in globalized urban centers. This growing disparity has intensified housing market instability and made it increasingly difficult for low- and middle-income households to access affordable housing. As a result, the issue of housing affordability became a widespread concern during this period, and when combined with accelerating population aging, it brought the question of elderly housing to the forefront of global academic inquiry.

Although interest in elderly residential issues has grown steadily, existing studies often concentrate on specific aspects such as post-discharge care services or barrier-free housing design. For instance, a demographic study from Germany revealed that only a small proportion of older adults reside in barrier-reduced housing even when mobility limitations are present [12], while a mixed-methods survey of occupational therapists in Australia emphasized the importance of specific design features such as step-free entrances and showers in supporting aging in place and post-discharge transitions [13].

Comprehensive analyses that integrate cross-national or cross-disciplinary perspectives remain relatively limited. Most

*Corresponding authors

existing studies are confined to single-country contexts or emphasize narrow thematic areas, which restricts their generalizability. In addition, there is a lack of integrative reviews that systematically map how different disciplines approach elderly residential issues, leaving the broader research landscape insufficiently explored. Therefore, this paper seeks to broaden the scope by employing a bibliometric approach to examine the overall research landscape on "elderly residential" topics. Through the analysis of key elements such as countries, disciplines, journals, and institutional affiliations, this study aims to identify global research patterns and publishing outlets valuable to prospective authors. Bibliometric analysis has proven to be an effective tool for quantitatively assessing scientific trends and knowledge structures within a research field, especially in emerging or interdisciplinary domains [14]. In the field of architecture, for instance, bibliometric techniques have been used to map trends in elderly-related design studies, revealing increasing attention to housing, accessibility, and aging-in-place themes [15]. Similarly, recent analyses of social support systems for older adults also demonstrate the utility of bibliometric methods in identifying evolving research focuses and policy implications [16].

This paper is structured as follows. Section II describes the data collection process, inclusion and exclusion criteria, and the analytical methods employed. Section III presents the main results, including country-level publication trends, disciplinary distribution, institutional contributions, and journal co-citation patterns. Section IV discusses the implications of these findings in the context of global aging, interdisciplinary integration, and policy relevance. Section V concludes the paper by summarizing the key insights and outlining directions for future research.

II. Materials and Methods

A. Data Collection and Filtration

We conducted a comprehensive literature search on the Web of Science Core Collection (WOSCC) from 2002-01-01 to 2021-12-31 [17]. The search was initiated on February 24, 2022. This database was chosen since it is one of the most well-known databases for conducting bibliometric analysis, currently covering 18,000 journals and classifying information into 256 subject categories [17], and has been widely employed in bibliometric studies across fields such as medical research, environmental science, information science, and engineering [16-18]. It provides structured citation networks and comprehensive reference tracing capabilities [19-21], which facilitate diverse bibliometric analyses using software tools. Therefore, the relevant bibliographic records on the topic of elderly residential were derived from WOSCC.

Inclusion and exclusion criteria: Only publications written in English were included to ensure consistency in text analysis. All document types indexed in WOSCC, including articles, reviews, conference proceedings, and book chapters, were initially retrieved. Duplicates were removed. Non-scholarly items such as editorials, corrections, and news items were excluded. Studies not directly related to elderly residential or housing for older adults were also excluded after a manual screening of titles and abstracts.

Search strategy: To maximize coverage, the search strategy was designed to capture a broad range of terms related to elderly residential research. First, synonyms for the target population were identified, including elderly and senior (with truncation "senior*" to capture both singular and plural forms). Second, synonyms for the core concept of residential housing were specified, including residential, housing for the elderly, and elderly housing. These terms were combined using Boolean operators to ensure sensitivity and precision. The final query string was: TS=[(elderly OR senior*) AND ("residential" OR "housing for the elderly" OR "elderly housing")] AND LA=(English) AND PY=(2002-2021). The language is restricted to English, and no restrictions in terms of document type, data category are used. Following the above criteria, "elderly residential" obtained 2,141 original records.

Analysis process: The retrieved bibliographic records were downloaded in plain text format with full records and cited references. Data preprocessing included removal of duplicates and standardization of author, institution, and journal names. Summary statistics, such as annual publication trends, country contributions, and research fields, were generated directly from WOSCC and Microsoft Excel. For network analyses, VOSviewer (version 1.6.17) was employed to map co-authorship, institutional collaboration, keyword co-occurrence, and journal co-citation networks. Nodes in the visualizations represent units such as authors, institutions, or journals, while links indicate the strength of their relationships. The built-in clustering algorithm grouped related items, and both network and density views were adopted to enhance interpretability.

B. Data Analysis

Fig. 1 illustrates the whole study process. The retrieved documents were downloaded from the Web of Science Core Collection and used as input data for the subsequent bibliometric analysis. Summary statistics, including annual publication trends, leading affiliations, and research areas, were extracted directly from the WOSCC platform. The analysis was performed using VOSviewer (version 1.6.17) for network visualization and Microsoft Excel for statistical summaries and charting. Microsoft Excel was used to generate multiple visualizations and summary tables, including country-level publication statistics, time-segmented distribution of top research fields, and bar charts illustrating the publication output of leading institutions. These visual elements provided a clear overview of the geographic and thematic distribution of research efforts in elderly residential. Basic data processing functions and built-in charting tools in Excel facilitated the organization and presentation of bibliometric indicators derived from the raw dataset [22].

VOSviewer specializes in creating maps based on bibliographic data, particularly useful for detecting co-authorship patterns, institutional collaboration networks, co-citation clusters, and keyword co-occurrence [23]. The software employs a distance-based mapping technique combined with a unified layout and clustering algorithm, allowing items with stronger relationships to be positioned closer together in the visual space. It is capable of processing large-scale bibliographic datasets while providing an intuitive interface for exploring

complex network structures. VOSviewer also offers several types of visualizations, including network, overlay, and density views, which support both qualitative interpretation and quantitative comparison of bibliometric connections.

This study utilized VOSviewer to perform three core bibliometric analyses: mapping international collaboration between countries, examining institutional co-authorship relationships, and identifying journal co-citation patterns. These analyses enabled a structured examination of collaborative patterns and the intellectual structure of the field. Nodes in these networks represent entities such as countries, institutions, or journals, while the links reflect the strength of their bibliometric relationships. The clustering algorithm embedded in VOSviewer further enables the grouping of closely related items, revealing underlying thematic structures within the dataset. Layout and density visualizations were adopted to enhance interpretability and support intuitive analysis of bibliometric connections.

III. Results

A. Countries

Table Ⅰ presents the scientific output of various countries and regions in the field of elderly residential research from 2002 to 2021. The United States ranked first with 379 publications (17.71% of the total), followed by Australia with 348 publications (16.26%) and China with 245 publications (11.45%). Together, these three countries contributed over 45% of the global research output, highlighting their central role in this field.

After normalizing by population size, Sweden showed the highest number of publications per million people (13.91), followed by Australia (13.55) and New Zealand (11.02), indicating that countries with relatively small populations but strong research intensity performed exceptionally well in terms of per capita output. Meanwhile, regions with high population density, such as Taiwan (6.51) and South Korea (5.17), also demonstrated active research performance, possibly reflecting heightened attention to elderly housing issues amid rapid urbanization and population aging.

In terms of collaboration networks, as illustrated in Fig. 1, North America, particularly the United States and Canada, maintained strong and frequent cooperation with Asian countries such as China, India, and South Korea. Australia, which ranks second in total publications, also showed close international collaboration, underscoring its key role in global elderly residential research. In contrast, countries like Germany, Italy, and Spain, despite having a moderate volume of publications, ranked relatively low in per capita output. This suggests that while aging is a global issue, national research investment and strategic focus vary considerably.

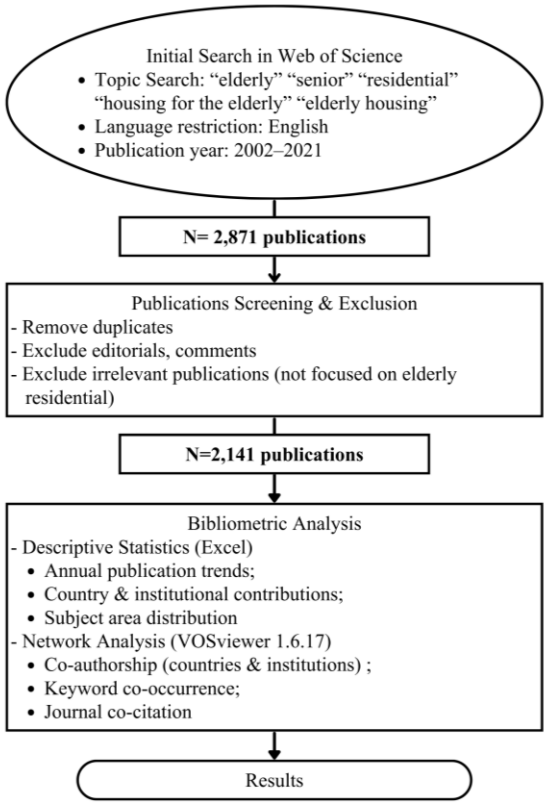


Fig. 1. Flowchart of the study process, including data collection, screening, and bibliometric analysis.

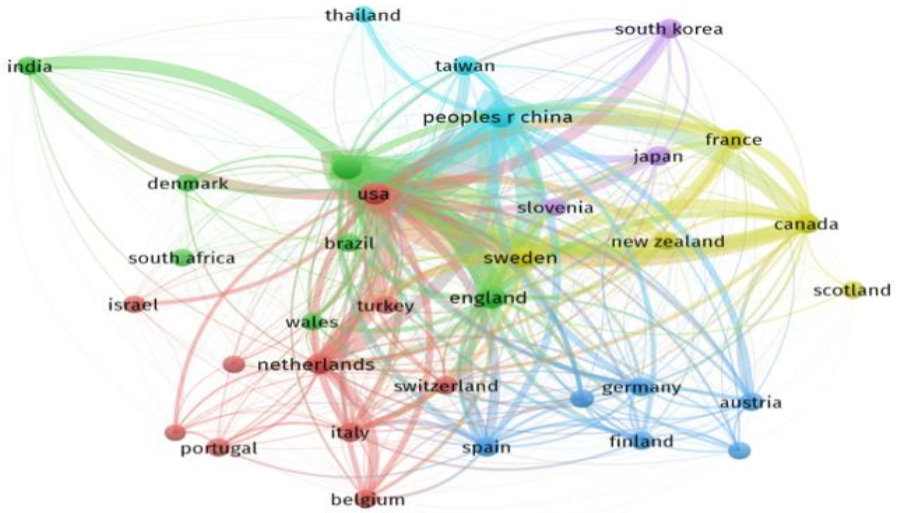
Fig. 2 visualizes the country-level collaboration network within the field of elderly residential research. Each node represents a country or region, with node size indicating the number of publications. The thickness of the connecting lines reflects the frequency of international research collaboration, while different colors represent clusters of countries that frequently co-author publications. The analysis reveals that the United States (USA) occupies the most central and prominent position in the network, indicating its leading role in both publication output and the breadth of global collaboration. China (People's Republic of China) and the United Kingdom (England) also demonstrate strong research productivity and maintain extensive links with a wide range of countries, highlighting their growing influence in this domain. A dense collaborative cluster is formed among several European countries, including Germany, Spain, Italy, the Netherlands, and Portugal, suggesting frequent intra-regional cooperation.

In parallel, the Asia-Pacific region, notably Japan, South Korea, Thailand, and Taiwan, is increasingly integrated into the global research landscape. Countries such as India, Canada, and Australia function as cross-regional connectors, facilitating scholarly exchange between Eastern and Western institutions.

TABLE I. COUNTRY-LEVEL RESEARCH OUTPUT ON ELDERLY RESIDENTIAL (2002–2021)

Countries and areas	# of Publication	% of total	Population in 2021(millions)	# of publications (per million population)	Land Area (10,000/m²)	Density (pop. per Land Area)	Rank (Pub.)	Rank (Pub. per million pop)	Rank (Density)
USA	379*	17.71*	332.47	1.14	983.40	0.34	1	13	10
AUSTRALIA	348*	16.26*	25.69	13.55	769.20	0.03	2	2	15
PEOPLE'S R CHINA	245*	11.45*	1412.6	0.17	959.70	1.47	3	15	7
ENGLAND	243	11.36	67.08	3.62	13.03	5.10	4	7	3
SWEDEN	144	6.73	10.35	13.91	45.03	0.23	5	1	12
NETHERLANDS	138	6.45	17.44	7.91	4.15	4.20	6	4	4
CANADA	117	5.47	38.01	3.08	998.50	0.04	7	8	14
GERMANY	91	4.25	83.24	1.09	35.74	2.33	8	14	5
ITALY	80	3.74	59.55	1.34	30.13	1.98	9	11	6
SPAIN	76	3.55	47.35	1.61	50.60	0.94	10	10	9
JAPAN	73	3.41	12.58	5.80	37.80	0.33	11	6	11
SOUTH KOREA	67	3.13	51.78	1.29	10.02	5.17	12	12	2
NEW ZEALAND	56	2.62	5.08	11.02	26.80	0.19	13	3	13
TAIWAN	51	2.38	23.57	2.16	3.62	6.51	14	9	1
FRANCE	50	2.34	67.39	7.42	54.39	1.24	15	5	8

*Note: * USA, Australia, and the People's Republic of China have been marked with an asterisk to indicate their position as the top three countries by number of publications in the field of elderly residential. # of Publication = Number of publications; % of total = Percentage of total publications; Population in 2021 (millions) = Population based on WHO 2021 statistics (in millions); # of publications (per million population) = Publications per one million people; Land Area (10,000/m²) = Land area in units of 10,000 square kilometers; Density (pop. per Land Area) = Population per unit of land area; Rank (Pub.) = Rank based on total publications; Rank (Pub. per million pop) = Rank based on publications per million population; Rank (Density) = Rank based on population density.*



Note: USA = United States; peoples r china = People's Republic of China; South Korea = Republic of Korea; Uk = United Kingdom ; UAE = United Arab Emirates.

Fig. 2. Country-level collaboration network in the field of elderly residential research (2002-2021).

B. Fields

Table Ⅱ illustrates the top five research fields in the domain of elderly residential across four consecutive five-year periods from 2002 to 2021, based on the Web of Science subject classification. Throughout all time intervals, Geriatrics Gerontology consistently ranked as the most prominent field, reflecting its foundational role in elderly residential research. Notably, the field reached its peak publication volume during 2012-2016 (251 articles), before declining slightly in the 2017-2021 period (195 articles).

Public, Environmental, and Occupational Health maintained its position as the second most represented field across all periods, with a steady increase in publication output from 44 articles (2002-2006) to 139 articles (2017-2021), highlighting the growing integration of environmental and public health perspectives in elderly housing research. Psychiatry remained a stable third-ranked field during the earlier periods, although it was later surpassed by Environmental Sciences and Gerontology in the last five years, indicating a disciplinary shift towards more environment- and aging-specific research contexts.

The emergence of Nursing, Internal Medicine, and Health Care Sciences and Services in earlier periods underscores the initial focus on clinical and institutional aspects of elderly. However, fields such as Environmental Studies and Gerontology gained prominence in the most recent period, reflecting an expansion of research into broader socio-environmental dimensions of aging.

TABLE II. TOP 5 RESEARCH FIELDS IN ELDERLY RESIDENTIAL BY FIVE-YEAR INTERVALS (2002-2021)

Rank	Fields (2002-2006)	# of Publication	Fields (2007-2011)	# of Publication	Fields (2012-2016)	# of Publication	Fields (2017-2021)	# of Publication
1	Geriatrics Gerontology[a]	183*	Geriatrics Gerontology[a]	229*	Geriatrics Gerontology[a]	251*	Geriatrics Gerontology[a]	195*
2	Public Environmental Occupational Health	44*	Public Environmental Occupational Health	54*	Public Environmental Occupational Health	88*	Public Environmental Occupational Health	139*
3	Psychiatry	39*	Psychiatry	48*	Psychiatry	44*	Environmental Sciences	104*
4	Health Care Sciences Services	22	Nursing	44	Medicine General Internal	42	Gerontology	86
5	Medicine General Internal	22	Medicine General Internal	25	Nursing	36	Environmental Studies	63

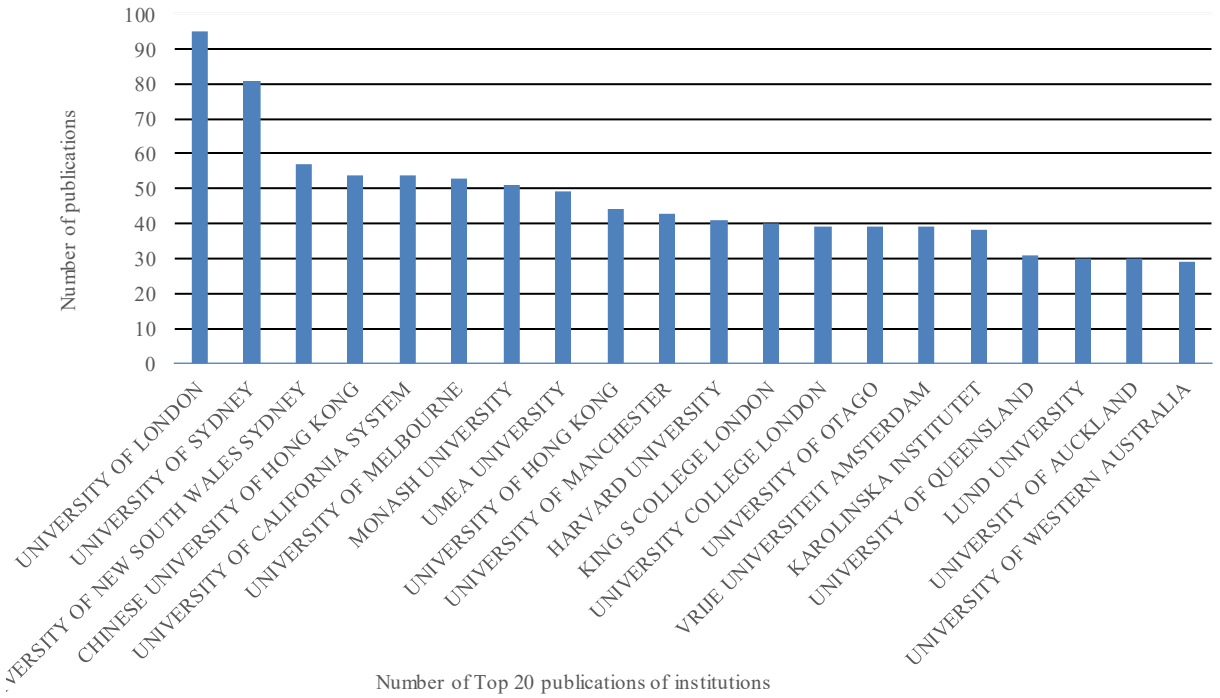
*Note: *indicates that the field ranked among the top three in the corresponding time. "Geriatrics Gerontologya" refers to the combined subject categories of "Geriatrics" and "Gerontology" in the Web of Science Core Collection (WOSCC).*

C. Institutions

Fig. 3 presents the publication output of the top 20 institutions contributing to the field of elderly residential research from 2002 to 2021. The University of London leads with the highest number of publications, reflecting its dominant academic position in this domain. It is followed by the University of Sydney and the University of New South Wales Sydney, both of which show high levels of scholarly productivity, underlining Australia's significant engagement in elderly residential topics.

The list also includes notable institutions from Hong Kong (e.g. Chinese University of Hong Kong), North America (e.g. University of California System, Harvard University), and Europe (e.g. University of Manchester, Karolinska Institutet). This wide distribution indicates that elderly residential research has attracted global attention and academic investment.

The recurring appearance of multiple institutions from countries like Australia and the UK suggests regional hubs of research excellence. The results highlight a trend where both English-speaking countries and regions with rapidly aging populations have contributed substantially to the knowledge base in this field.



Note: Each bar represents the number of publications produced by each institution related to elderly residential research during the period 2002-2021.

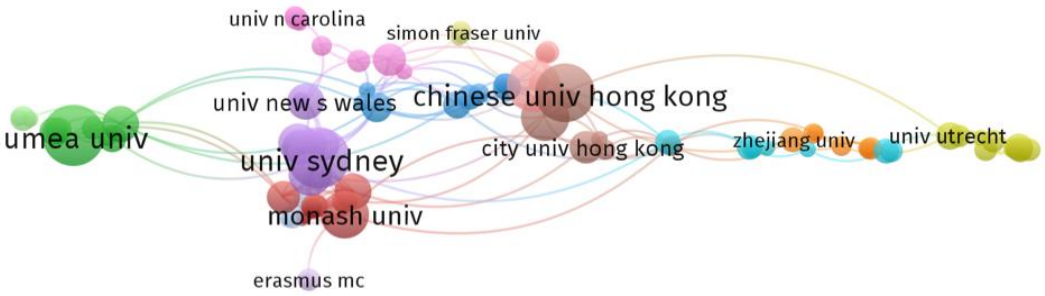
Fig. 3. Number of publications of the top 20 most productive institutions in elderly residential research (2002-2021).

Fig. 4 displays the institutional co-authorship network in the field of elderly residential research. Each node represents a university or research institution, with node size proportional to its total publication output. The thickness of the connecting lines indicates the frequency of co-authored publications, reflecting the intensity of academic collaboration.

The network reveals a high degree of international cooperation. Central institutions include Umeå University, University of Sydney, Monash University, and the Chinese University of Hong Kong, all of which demonstrate high publication output and strong collaborative ties. Regional clustering is evident, particularly among institutions in Australia and Hong Kong, where close intra-regional collaboration is observed. Several other Australian institutions also exhibit substantial academic activity in this domain. The University of New South Wales ranks third in total publications and shows a strong pattern of international collaboration, while the University of Melbourne ranks within the top 10 and maintains frequent co-authorships with leading global research centers.

In the United Kingdom, both the University of London and the University of Manchester are among the top 10 most productive institutions, highlighting the country's advanced position in this field. Nordic countries also contribute significantly, with Umeå University (Sweden) emerging as one of the most influential institutions in terms of both output and connectivity.

Furthermore, the presence of institutions such as Zhejiang University (China) and Utrecht University (Netherlands) reflects growing global engagement, particularly from Asia and continental Europe. The observed collaboration structure indicates a distributed, multi-centered international research network, supporting the expansion of scholarly efforts in elderly residential.



Note: umea univ = Umeå University; univ n carolina = University of North Carolina; univ new s wales = University of New South Wales; univ sydney = University of Sydney; monash univ = Monash University; chinese univ hong kong = The Chinese University of Hong Kong; city univ hong kong = City University of Hong Kong; simon fraser univ = Simon Fraser University; zhejiang univ = Zhejiang University; univ utrecht = Utrecht University; erasmus mc = Erasmus Medical Center.

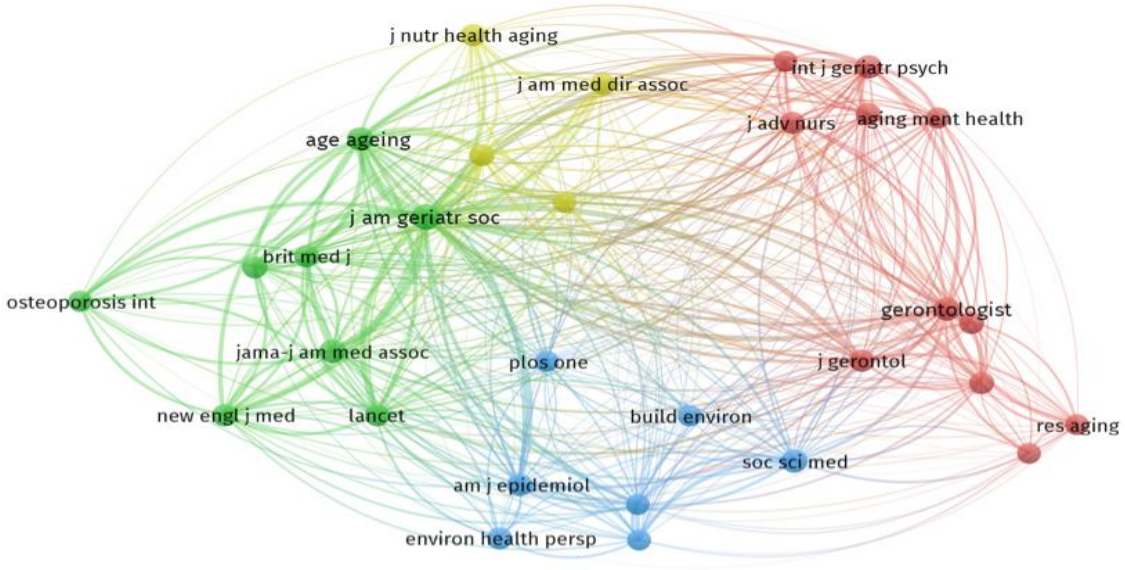
Fig. 4. Institutional co-authorship network in the field of elderly residential research.

D. Journals

Fig. 5 presents the journal co-citation network within the domain of elderly residential research. Each node represents an academic journal, with the node size corresponding to its co-citation frequency, and colors denoting different thematic clusters. Links between nodes indicate the strength of co-citation relationships, reflecting the underlying intellectual structure and disciplinary interconnections in the field.

The network reveals several prominent clusters. Core gerontology journals, Journal of the American Geriatrics Society, The Gerontologist, and Journals of Gerontology, occupy central positions, indicating their foundational role in shaping aging-related scholarship. Another cluster comprises interdisciplinary journals such as Social Science & Medicine, Environmental Health Perspectives, and Building and Environment, suggesting close linkages between aging, environmental health, and the social sciences. Additionally, high-impact clinical journals, including JAMA, The Lancet, and New England Journal of Medicine, are also represented, underscoring the relevance of aging issues in broader public health and medical research.

In terms of publication volume, Age and Ageing was the most productive journal with 61 articles, followed by International Journal of Environmental Research and Public Health with 44, and Journal of the American Geriatrics Society with 32. These publication patterns align with the journals' disciplinary classifications: Age and Ageing and Journal of the American Geriatrics Society fall under Geriatrics & Gerontology, while International Journal of Environmental Research and Public Health is categorized under both Environmental Sciences and Public, Environmental & Occupational Health.



Note: j nutr health aging = The Journal of Nutrition, Health and Aging; j am med dir assoc = Journal of the American Medical Directors Association; age ageing = Age and Ageing; j am geriatr soc = Journal of the American Geriatrics Society; brit med j = The British Medical Journal (BMJ); jama-j am med assoc = Journal of the American Medical Association (JAMA); new engl j med = The New England Journal of Medicine; lancet = The Lancet; osteoporosis int = Osteoporosis International; plos one = PLOS ONE; am j epidemiol = American Journal of Epidemiology; environ health persp = Environmental Health Perspectives; build environ = Building and Environment; soc sci med = Social Science & Medicine; int j geriatr psych = International Journal of Geriatric Psychiatry; j adv nurs = Journal of Advanced Nursing; aging ment health = Aging & Mental Health; gerontologist = The Gerontologist; j gerontol = Journals of Gerontology; res aging = Research on Aging.

Fig. 5. Journal co-citation network in the field of elderly residential research.

IV. Discussion

A. Comparison with Related Research

Recent evidence synthesizes how the built environment supports ageing in place through urban planning, physical accessibility, and adaptable housing design, while also noting inconsistencies in reported effects on cognitive outcomes and well-being [24]. One systematic review in this area calls for integrated approaches that link housing, health, and planning and for clearer operationalization of environmental exposures and outcomes [25]. Qualitative work from Sweden adds demand-side insight into how active older adults actually use neighborhoods [26]. Interviews grounded in time-use diaries show that proximity to everyday amenities, walkable public spaces, and opportunities for spontaneous social interaction structure daily practices and support autonomy, with some gender differences and variation between central and peripheral locations [26]. These findings underscore the importance of neighborhood-scale design for health and participation. A complementary strand links senior-friendly housing to the silver economy [27]. A recent scoping review emphasizes universal design and supportive environments as means to sustain autonomy and reduce dependency while creating economic opportunities, illustrating how housing policy can align with broader development agendas in rapidly ageing settings [27].

Although this study systematically mapped the development of global elderly housing research between 2002 and 2021 through bibliometric methods and provided a macro-level perspective on cross-national comparisons, disciplinary evolution, and collaboration networks, several limitations remain. In contrast to the systematic review [25], our analysis did not examine in detail how specific built environment factors influence health outcomes, and therefore offers limited insights into underlying mechanisms. Compared with the qualitative work of Vilhelmson [26], our study relies primarily on English-language publications indexed in the Web of Science, which constrains the ability to capture older adults' lived experiences and local contextual differences. Relative to the scoping review [27], which links senior-friendly housing to the silver economy and policy frameworks, our work has not fully addressed the intersections between housing, economic development, and industrial policy. Thus, the main contribution of this study lies in delineating the knowledge structure and interdisciplinary trends of the field, while further research is required to advance understanding at the levels of health impacts, individual experiences, and economic–policy integration [28].

B. Responding to Global Aging Through Residential Research

The ongoing demographic transformation marked by global aging has substantially reshaped housing-related academic discourse. As populations' age and urban centers experience real estate intensification, the housing needs of older adults become increasingly complex. Recent studies have linked rising housing prices with growing inequality in housing access, particularly among low-income elderly residents [29]. As affordability declines, aging populations, especially those living alone or on fixed incomes, are disproportionately vulnerable to housing instability [30]. This socio-economic backdrop is reflected in scholarly publication trends. Between 2002 and 2021, the field of elderly residential research expanded significantly, with countries such as the United States, Australia, and China contributing over 45% of global publications. When normalized

by population, smaller nations like Sweden, New Zealand, and the Netherlands demonstrated exceptionally high research intensity. This pattern suggests that both demographic pressure and housing market volatility have acted as research catalysts, especially in high-density or rapidly urbanizing regions.

The geographic concentration of output, especially in nations facing acute housing affordability issues, indicates a clear response to national policy priorities and market realities. For example, regions such as Taiwan and South Korea demonstrated increasing academic interest in elderly housing, likely shaped by urban crowding, transport inequality, and land scarcity. A recent study in South Korea confirmed that accessibility to health and aged care infrastructure is significantly lower in urban peripheries, disproportionately affecting older populations [31]. This intersection of spatial inequality and demographic stress has helped prioritize elderly housing as a key research and policy agenda across disciplines.

C. Changing Research Fields and Emerging Interdisciplinary Trends

Geriatrics and gerontology continue to form the empirical backbone of elderly residential studies, particularly in addressing healthcare accessibility and long-term care provision. Yet, a notable disciplinary shift has occurred over the past decade toward the inclusion of environmental sciences and urban studies. These fields now contribute substantially to the discourse on aging-in-place strategies and the role of built environments in promoting active aging [32]. Recent studies in urban planning have emphasized the role of neighborhood walkability, greenery, and social infrastructure in supporting older adults' autonomy [33]. Environmental determinants, such as air quality, spatial safety, and public transport accessibility, are increasingly framed as health enablers for aging populations [34]. This expansion signals a growing consensus that elderly housing research must be interdisciplinary to address the full spectrum of residential quality and well-being.

D. Deepening Focus on Environmental Sciences and Urban Studies

The role of environmental factors and urban design in shaping older adults' well-being has gained substantial scholarly attention. Concepts such as green space aesthetics, walkability, and neighborhood connectivity are now central to research on aging in place [35]. Empirical tools have been developed to assess age-friendliness in residential environments, such as multi-dimensional frameworks measuring accessibility, public service proximity, and environmental quality.

Design-based and participatory research approaches have also emerged. In Prague, for example, user-centered design and digital innovation are applied to tailor home environments for older users [36]. Studies in Italy emphasize the integration of shared green infrastructure, public seating, and intergenerational community planning as ways to promote psychological and physical well-being among seniors [37]. Walkability, including the presence of pedestrian infrastructure and nearby amenities, plays a crucial role in encouraging walking and mobility among older adults [38].

E. Regional Leadership and Policy-Driven Research Growth

Australia and Hong Kong stand out as leading contributors to elderly residential research, driven by targeted policy reforms and high research intensity. Australia's aged care strategies, such as the implementation of Home Care Packages (HCP) and Residential Aged Care (RAC), have stimulated academic studies on service design and housing adequacy [39]. In Hong Kong, the government has integrated welfare instruments such as housing subsidies and community living allowances to enhance older adults' residential stability. These efforts have also generated substantial institutional collaborations between local universities and public agencies [40]. Additionally, elderly housing policy in Hong Kong has been shaped by demographic projections and the urgent need to prevent future institutional care deficits. While direct strategies to reduce institutionalization have been limited, policy responses, such as public housing redevelopment supporting aging in place, reflect a growing recognition of this challenge. As noted by Mesthrige and Cheung (2020), such initiatives not only aim to promote residential autonomy among older adults but also encourage academic engagement in evaluating housing adequacy and long-term care alternatives [41]. This evolving policy-academic feedback loop positions Hong Kong as a key case for understanding how institutional design and aging policy can co-produce robust research ecosystems.

F. Strengths and Limitations

This study employs bibliometric methods to synthesize global research trends in elderly housing over 20 years. The strength of this approach lies in its capacity to capture interdisciplinary growth, highlight collaboration networks, and quantify shifts in publication output. Furthermore, combining subject category analysis with institutional and regional metrics provides a more holistic understanding of where and how knowledge on elderly residential is produced [42].

Nonetheless, limitations must be acknowledged. The analysis relies exclusively on the Web of Science Core Collection, potentially excluding valuable research indexed in Scopus, PubMed, or regional databases. Furthermore, restricting the review to English-language publications likely underrepresents elderly housing research in Latin America, East Asia, and non-Anglophone Europe [14]. Third, citation-based metrics used in bibliometric studies often favor older, well-established research while undervaluing recent or practice-oriented studies that may not yet have accumulated citations [43]. This limitation is particularly salient in emerging interdisciplinary fields, like elderly housing, where impactful policy papers and architectural studies may not circulate in highly cited scientific journals.

G. Future Research Directions

The increasing complexity of elderly housing issues calls for expanded and more nuanced research agendas. While recent years have seen growing interdisciplinary attention to aging in place, urban livability, and health equity, several promising entry points remain underexplored and merit future investigation.

First, researchers should prioritize comparative policy studies to examine how different welfare regimes, urban development models, and long-term care systems influence

elderly housing access and design. National policies can profoundly shape housing affordability and support structures, particularly in rapidly aging but economically diverse regions [44]. Second, future research should expand on technology-enabled living environments for older adults, particularly examining the integration of smart home systems, telehealth, and ambient sensors. However, more work is needed to assess ethical concerns, user acceptance, and long-term cost-effectiveness [45]. Third, methodological innovation is essential. Researchers should adopt mixed-method designs, longitudinal datasets, and community-based participatory approaches to capture the lived experiences of aging populations in different contexts. This would enrich quantitative bibliometric insights with qualitative depth and social meaning [46].

Lastly, considering escalating environmental challenges, climate-resilient senior housing should become a core research agenda. Studies should evaluate how extreme weather events, urban heat islands, and infrastructure breakdowns uniquely impact vulnerable older populations, and how housing design can mitigate such risks [47].

V. Conclusion

This bibliometric analysis reveals a rapidly expanding and increasingly interdisciplinary field of research centered on elderly residential from 2002 to 2021. With contributions led by countries such as the United States, Australia, and China, and strong per capita output from Sweden and New Zealand, the study highlights a globally shared academic response to the dual challenges of population aging and housing affordability. The analysis further demonstrates a clear disciplinary shift from traditional geriatrics and nursing toward environmental sciences, urban planning, and public health, underscoring the structural interdisciplinarity of the field. Furthermore, international and institutional collaboration networks demonstrate growing global engagement and policy relevance. By mapping publication trends, research clusters, and collaborative patterns, this study not only synthesizes two decades of scholarly development but also lays a foundational framework to guide future inquiries into inclusive, sustainable, and health-promoting housing for aging populations. Journal co-citation patterns also reveal that core gerontology journals are closely linked with outlets in environmental health and the social sciences, indicating the cross-field integration of knowledge.

Overall, these findings contribute in three ways. First, they delineate the global knowledge structure of elderly housing and identify leading countries, institutions, and journals. Second, they highlight policy-relevant fronts such as accessibility, aging in place, and community-level design where evidence has accumulated. Third, they provide a reproducible baseline for future longitudinal updates of the field. Practical implications suggest that inclusive housing strategies should be co-designed across health and planning systems, with greater attention to accessibility retrofits and community services that delay institutionalization. Future research should expand comparative policy studies, apply mixed-method designs, and address climate resilience in senior housing to meet the challenges of aging populations.

References

[1] L. W. Roberts, "Preparing for the needs of our aging population," Acad Med (San Franc), vol. 99, no. 2, pp. 123-125, February 2024.

[2] W. Bank. "Population ages 65 and above (% of total population)." World Bank. https://data.worldbank.org/indicator/SP.POP.65UP.TO.ZS (accessed June, 2024).

[3] D. Gu, K. Andreev, and M. E. Dupre, "Major trends in population growth around the world," China CDC Wkly, vol. 3, no. 28, pp. 604-613, July 2021.

[4] F. Paschaline, R. Prastita, and E. Mega, "Japan's aging society: A challenge to Japan's diversity & social inclusion," J. Transform. Glob., vol. 10, no. 3, pp. 20-34, July 2023.

[5] C. F. Payne, K. Q. Xu, and J. Y.-H. Aw, "Healthy longevity in the Asia–Pacific: a cross-national population-based modeling study," Am. J. Epidemiol., vol. 00, no. 00, pp. 1-9, July 2025.

[6] J. Sung, O. M. Dean, R. C. Saunders, J. Sheriff, and A. Unuigbe, "Affordability challenges for older adults in the nongroup market," AARP Public Policy Institute, Washington, DC, Report March 2021.

[7] A. Molassiotis, S. W. Kwok, A. Y. Leung, and S. Tyrovolas, "Associations between sociodemographic factors, health spending, disease burden, and life expectancy of older adults (70+ years old) in 22 countries in the Western Pacific Region, 1995–2019: estimates from the Global Burden of Disease (GBD) Study 2019," Geroscience, vol. 44, no. 2, pp. 925-951, January 2022.

[8] A. Melecky and D. Paksi, "Drivers of European housing prices in the new millennium: demand, financial, and supply determinants," Empirica, vol. 51, no. 3, pp. 731-753, April 2024.

[9] R. Greenaway-McGrevy and P. C. Phillips, "House prices and affordability," N. Z. Econ. Pap., vol. 55, no. 1, pp. 1-6, March 2021.

[10] J. Kim, "Aging, housing prices, and young adults' homeownership," Cities, vol. 149, no. 104914, February 2024.

[11] Y. Lee, P. A. Kemp, and V. J. Reina, "Drivers of housing (un) affordability in the advanced economies: A review and new evidence," Hous. Stud., vol. 37, no. 10, pp. 1739-1752, September 2022.

[12] S. Nowossadeck, L. Romeu Gordo, and A. Lozano Alcántara, "Mobility restriction and barrier-reduced housing among people aged 65 or older in Germany: Do those who need it live in barrier-reduced residences?" Front. Public Health, vol. 11, pp. 1-7, April 2023.

[13] C. Wellecke, K. D'Cruz, D. Winkler, J. Douglas, I. Goodwin, and E. Davis, et al., "Accessible design features and home modifications to improve physical housing accessibility: A mixed-methods survey of occupational therapists," Disabil. Health J., vol. 15, no. 3, February 2022.

[14] E. Seo and S. Lee, "Implications of aging in place in the context of the residential environment: bibliometric analysis and literature review," Int. J. Environ. Res. Public Health, vol. 20, no. 20, October 2023.

[15] R. M. Sarı and H. Eyüboğlu, "Reading the older people researches in architecture through bibliometric analysis (1975-2022)," Period. Polytech. Archit., vol. 54, no. 3, pp. 215-228, December 2023.

[16] M. S. Hossen and H. M. Pauzi, "Bibliometric analysis of social support for the older adults," Ageing Int., vol. 50, no. 1, pp. 1-24, October 2025.

[17] V. K. Singh, P. Singh, M. Karmakar, J. Leta, and P. Mayr, "The journal coverage of Web of Science, Scopus and Dimensions: A comparative analysis," Scientometrics, vol. 126, no. 6, pp. 5113-5142, March 2021.

[18] J. Aldás-Onofre and B. Cordero, "Bibliometric analysis of Web of Science database STEM fields in engineering and mathematics. Ecuador's case study," International Conference on Applied Technology, pp. 255-270, January 2022.

[19] M. Han, F. Yang, and H. Sun, "A bibliometric and visualized analysis of research progress and frontiers on health effects caused by PM2. 5," Environ. Sci. Pollut. Res., vol. 28, no. 24, pp. 30595-30612, April 2021.

[20] S. Jiang, D. Ma, X. Tan, M. Yang, Q. Jiao, and L. Xu, "Bibliometric analysis of the current status and trends on medical hyperspectral imaging," Front. Med., vol. 10, no. 1235955, September 2023.

[21] X. Huang, Z. Yang, J. Zhang, R. Wang, J. Fan, and H. Zhang, et al., "A bibliometric analysis based on web of science: current perspectives and potential trends of SMAD7 in oncology," Front. Cell Dev. Biol., vol. 9, no. 712732, February 2022.

[22] E. Hossain, Introduction to Microsoft Excel (Excel Crash Course for Engineers). Cham: Springer. pp. 1-18, April 2021.

[23] U. A. Bukar, M. S. Sayeed, S. F. A. Razak, S. Yogarayan, O. A. Amodu, and R. A. R. Mahmood, "A method for analyzing text using VOSviewer," MethodsX, vol. 11, no. 102339, August 2023.

[24] C. Heller, M. Haak, S. M. Schmidt, C. Chiatti, L. Ekstam, and M. H. Nilsson, et al., "The relationship between physical housing characteristics, housing accessibility and different aspects of health among community-dwelling older people: A systematic review," J. Aging Health, vol. 36, no. 1-2, pp. 120-132, June 2024.

[25] J. Molinsky and A. Forsyth, "Climate change, aging, and well-being: how residential setting matters," Hous. Policy Debate, vol. 33, no. 5, pp. 1029–1054, August 2023.

[26] A. García Sánchez and A. Torres Barchino, "The influence of the built environment on the quality of life of urban older adults aging in place: a scoping review," J Aging Environ., vol. 38, no. 4, pp. 398-424, June 2024.

[27] G. Richards, J. Carpenter, E. Okpalugo, D. Howard, and C. Heneghan, "Assessing housing exposures and interventions that impact healthy cities: a systematic overview of reviews," Perspect. Public Health, vol. 145, no. 5, pp. 272-276, September 2023.

[28] B. Vilhelmson, E. Thulin, and A. Gil Solá, "Active aging in place: neighborhood practices and preferences among urban older adults in Sweden," Urban Geogr., vol. 46, no. 6, pp. 1-20, July 2025.

[29] W. Wang and N. J. Durst, "Planning for active aging: Exploring housing preferences of elderly populations in the United States," J. Hous. Built Environ., vol. 38, no. 2, pp. 795-809, August 2023.

[30] G.-R. Park, "Housing Hardship Among Older Adults Living Alone: The Protective Roles of Aging and Housing Policies," J. Appl. Gerontol., vol. 44, no. 7, pp. 1022-1029, October 2025.

[31] S. Lee, J. Im, and K. Cho, "Understanding spatial inequalities and stratification in transportation accessibility to social infrastructures in South Korea: multi-dimensional planning insights," Sci. Rep., vol. 14, no. 1, August 2024.

[32] B. Xia, Q. Chen, L. Buys, C. Susilawati, and R. Drogemuller, "Impact of the built environment on ageing in place: A systematic overview of reviews," Buildings, vol. 14, no. 8, July 2024.

[33] C.-Y. Yu and B. Wang, "Influence of neighborhood walkability on older adults' walking behavior, health, and social connections in third places," in "TRANSPORT FINDINGS," Report October 2024.

[34] A. Segura Cardona, D. Cardona Arango, A. Segura Cardona, C. Robledo Marín, and D. Muñoz Rodríguez, "Friendly residential environments that generate autonomy in older persons," Int. J. Environ. Res. Public Health, vol. 20, no. 1, December 2022.

[35] A. Mahmood, R. Patille, E. Lam, D. J. Mora, S. Gurung, and G. Bookmyer, et al., "Aging in the right place for older adults experiencing housing insecurity: An environmental assessment of temporary housing program," Int. J. Environ. Res. Public Health, vol. 19, no. 22, November 2022.

[36] J. M. BLažEK, T. H. SaMEc, P. Kubala, and V. Orcígr, "Participativní bydlení: nová konceptualizace demokratických forem bydlení v České republice," Sociol. Cas., vol. 59, no. 2, pp. 173-195, June 2023.

[37] L. Carrera, "The elderly and the right to an active aging: the strategy of social cohousing to counteract relational poverty," Front. Sociol., vol. 9, no. 1447614, December 2024.

[38] F. Herbolsheimer, A. Mahmood, Y. L. Michael, and H. Chaudhury, "Everyday walking among older adults and the neighborhood built environment: A comparison between two cities in North America," Front. Public Health, vol. 8, no. 564533, December 2020.

[39] V. Cornell and J. Ratcliffe, "Optimizing in-home care services to avoid residential aged care admission: an australian pilot study," J. Palliat. Care, vol. 36, no. 3, pp. 168-174, March 2021.

[40] Q. K. Qian, W. K. Ho, W. M. Jayantha, E. H. Chan, and Y. Xu, "Aging-in-Place and home modifications for urban regeneration," Land, vol. 11, no. 11, November 2022.

[41] J. W. Mesthrige and S. L. Cheung, "Critical evaluation of 'ageing in place' in redeveloped public rental housing estates in Hong Kong," Ageing Soc., vol. 40, no. 9, pp. 2006–2039, April 2020.

[42] M.-L. Li, J. Allen, V. Plummer, and Y.-L. Dai, "Comparison of research hotspots and trends in long-term care for the elderly between China and Australia: a bibliometric analysis†," Front. Nurs., vol. 9, no. 4, pp. 421–430, December 2022.

[43] M. Mehregan, "Scientific journals must be alert to potential manipulation in citations and referencing," Res. Ethics, vol. 18, no. 2, pp. 163-168, March 2022.

[44] Y. Geva and M. Siemiatycki, "Real estate for social purpose: varieties of entrepreneurialism in Toronto's non-profit housing sector," Int. J. Hous. Policy, pp. 1-20, August 2024.

[45] A. Vrančić, H. Zadravec, and T. Orehovački, "The role of smart homes in providing care for older adults: A systematic literature review from 2010 to 2023," Smart Cities, vol. 7, no. 4, pp. 1502–1550, June 2024.

[46] K. R. Ramamurthy, R; Divyasthri, N; Arjunan, R, "Senior-Friendly housing and the silver economy: a scoping review," J. Popul. Ageing, August 2025.

[47] M. S. Hossen and H. M. Pauzi, "Bibliometric analysis of social support for the older adults," Ageing Int., vol. 50, no. 1, pp. 1-24, October 2025.

Strategic Decision Support in Financial Management Using Deep Learning-Based Stock Price Prediction Models

Layth Almahadeen[1], Dr. Chinnapareddy Venkata Krishna Reddy[2], Roopa Traisa[3],
Mukhamadiev Sanjar Isoevich[4], Lavanya Kongala[5], Dr Janvi Anand Rathi[6], Dr Revati Ramrao Rautrao[7]
Department of Financial and Administrative Sciences, Al-Balqa' Applied University, Jordan[1]
Associate Professor, Department of Management Studies, ANURAG Engineering College (Autonomous), Suryapet, Telangana, India[2]
Department of Management-School of Management-UG, JAIN (Deemed to be University), Bangalore, Karnataka, India[3]
Department of Artificial Intelligence, Tashkent State University of Economics, Uzbekistan[4]
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., Andhra Pradesh - 522302, India[5]
Assistant Professor, Department of Business Administration, Wainganga College of Engineering and Management (Autonomous), Nagpur, Maharashtra, India[6]
Associate Professor, Department of Management, Dr. D. Y. Patil B-School Pune, Maharashtra, India

***Abstract*—The Strategic decision support creates a strong and smart decision support system for financial management through the correct forecast of stock prices with deep learning. Statistical models and shallow machine learning techniques tend to be ineffective in modeling the nonlinear relationships, sequential interdependencies, and time-dependent volatility typical of financial data; consequently, poor prediction quality and untrustworthy investment choices. To overcome these constraints, introduce a new hybrid deep learning architecture based on the Temporal Fusion Transformer (TFT) combined with Bidirectional Long Short-Term Memory (BiLSTM) networks. The proposed hybrid model is expected to complement time-series forecasting by simultaneously utilizing attention mechanisms for explainability and sequence memory functions for richer temporal understanding. The model is trained and tested with the publicly available Stock Market Dataset on Kaggle, which includes stock history from various companies. The whole process is carried out on the Python platform using TensorFlow along with relevant libraries for data preprocessing, feature scaling, and model training. The new TFT-BiLSTM model surpasses conventional models through an accuracy level of 93.4% and an F1-score of 94.2%, demonstrating its precision and generalization power. The system provides strategic benefits in financial planning and risk management. Financial analysts, investors, fintech, and portfolio managers may take advantage of our prediction system to make rational buy/sell judgments, minimize risks, and maximize asset allocations. By synthesizing state-of-the-art deep learning models and public financial data, our framework illustrates that accurate stock price prediction can be an effective mechanism for supporting decision-making in financial markets.**

Keywords*—Stock price forecasting; deep learning; temporal fusion transformer; financial decision support; BiLSTM

I. Introduction

Stock price prediction is a popular problem of considerable practical interest in investment, risk management, and strategic asset allocation across financial markets. Early approaches were dominated by traditional econometric models like autoregressive integrated moving average [1], generalized autoregressive conditional heteroskedasticity and Kalman filters focusing on predicting price changes by analyzing historical price and volatility patterns [2]. Such models, though, are frequently plagued by strong assumptions such as linearity, stationarity, and Gaussian errors, restricting their power to reflect the inherently nonlinear and chaotic characteristics of actual financial time series. As markets develop under the forces of high-frequency trading, macroeconomic trends, and intricate cross-asset dynamics, [3] it is critical to utilize richer market behavior representations. Therefore, more flexible and stronger models able to learn adaptively from data are essential to enhance prediction accuracy and usefulness.

Over the last decade, there has been growing use of deep learning methods, "recurrent neural networks (RNNs), long short-term memory networks (LSTMs), gated recurrent units (GRUs), and convolutional neural networks (CNNs)", to predict stock prices. These models are more effective in capturing nonlinear relationships and temporal dependencies in historical price data[4]. Variants like BiLSTM examine both forward and backward sequence context, and attention mechanisms in Transformers emphasize significant input times [5]. However, these approaches have limitations: RNN-based models can lose long-range dependencies or have vanishing gradients; Transformers are potentially data-inefficient and pure LSTM models are not interpretable [6]. Hybrid solutions have been suggested, e.g. CNN-LSTM or Attention-LSTM, in order to leverage strengths, but fundamental shortcomings still persist in integrating seamlessly interpretable attention with [7] sequential bi-directional recurrence without sacrificing accuracy and computational tractability.

In this research, a hybrid deep learning architecture consisting of a TFT and a Bidirectional LSTM (BiLSTM) to predict next-day stock closing prices is presented. Our method

utilizes TFT's strong attention and feature selection capacity to automatically recognize salient features at every time step, while BiLSTM captures sequence context from past as well as future information windows. They utilize the publicly accessible "Stock Market Dataset" on Kaggle, which provides daily OHLC, volume, adjusted close, and ticker symbols of several US-listed stocks for the last two years. This data set offers a modern and comprehensive source for model training and testing. Through the emphasis on recent data, our method adjusts to prevailing market tendencies. The combined model is trained on sliding-window sequences to forecast future closing prices, with a goal of surpassing traditional baselines according to MAE, RMSE, and R^2.

- Established the research goal by seeking to enhance stock price prediction accuracy by incorporating attention-based temporal models with bidirectional sequence learning.
- Optimized a hybrid TFT-BiLSTM architecture. Integrated TFT's transparent attention mechanisms and BiLSTM's two-way sequence modeling for a better understanding of temporal context.
- Used the stock market dataset. Exploited a diverse, recent, and rich two-year daily stock price dataset to train and test the hybrid model.
- Accomplished better performance. Achieved higher accuracy compared to baseline models, with lower MAE and RMSE, and greater R^2 in numerous tickers.

A. Motivation of the Study

In the fast-moving financial era, precise stock market prediction is a necessity for investors, fund managers, and institutions wanting to reduce risk and maximize return. Conventional models like ARIMA or linear regression models are not in a position to capture the nonlinear, dynamic, and time-dependent pattern of stock prices. With increasing advances in artificial intelligence and deep learning, there is a greater incentive to investigate more advanced ways that are capable of drawing out implicit patterns and long-term relationships from abundant sources of historical data. The incentive of this research is bridging the gap between raw financial information and actionable strategic choices by utilizing hybrid deep learning frameworks. Merging Temporal Fusion Transformers (TFT) with BiLSTM enables us to perform beyond mere prediction toward developing a smart, explainable, and scalable decision support system. This study hopes to enhance the accuracy of prediction while bringing real-world advantages in financial planning and portfolio management.

B. Significance of the Study

This research has high value in the confluence of stock market forecasting and artificial intelligence by providing a real-world, high-performance solution for forecasting stock market trends. Its value comes from the fact that the proposed TFT-BiLSTM model can beat conventional and current deep learning approaches by successfully learning to leverage complex sequential data. The determined by various variables over time, a hybrid model integrating temporal attention and memory mechanisms may provide higher accuracy and interpretability. The research is not only beneficial to academic researchers but also to industry professionals such as investors, stock traders, and data-driven fintech firms by providing a forecasting tool that facilitates strategic decision-making. By employing an actual-world dataset and running the solution using Python, the model is reproducible, scalable, and ready for use in trading systems or financial analytics platforms. All in all, this study contributes to smarter, AI-enabled financial planning in more volatile and competitive markets.

The rest of this paper follows the following structure: Section II, Related Works, reviews previous literature on stock prediction with both conventional and deep learning approaches. Section III, Problem Statement, mathematically defines the forecasting problem and model goals. Section IV, Methodology, outlines data collection, preprocessing processes, and our TFT-BiLSTM hybrid model structure. Section V, Results, reports experimental conditions, performance measures, and visual inspections of the predictions. Section VI concludes by summarizing findings and stating future directions of work.

II. Related Works

M. Elhoseny [8] et al., proposed the AWOA-DL model to enhance precision in credit risk forecasting and financial distress prediction. The model combines a deep neural network (multilayer perceptron) with an Adaptive Whale Optimization Algorithm (AWOA) for hyperparameter tuning, improving prediction accuracy. The model is run in three phases: preprocessing, tuning, and prediction. Its strength lies in high predictive performance (95.8%) and applicability across diverse datasets. However, its computational complexity limits scalability and real-time deployment in high-frequency financial systems or organizations with limited infrastructure. D. K. Nguyen [9] et al. examined the convergence and influence of Big Data, AI, and ML in transforming the fintech sector. It uses a multi-dimensional descriptive analysis approach to build and test a framework demonstrating their synergistic relationship in financial services. The benefit comes with the elucidation of how these technologies enable digital finance and redefine the role of the data scientist. Yet it also deals with drawbacks like ethical implications, government regulations, and risks of abuse or overreliance on automated decision-making.

A. Mehra [10] et al., introduced enrich intricate decision-making through the neural networks' pattern recognition power. A rule-based mechanism for symbolic reasoning and a neural network for deep learning are utilized in the suggested hybrid model, facilitating interaction within a single architecture. The benefits are enhanced accuracy, interpretability, and scalability in areas such as healthcare and finance. Nonetheless, difficulties lie in architectural complexity and possible integration challenges between structured and unstructured data processing elements. K. Li [11] et al., proposed measuring corporate culture using word embeddings to score five cultural values for firm-year observations (2001–2018). Advantages include a data-driven, scalable approach to assessing corporate culture and uncovering its link to performance, risk, and executive behavior. However, limitations include potential context misinterpretation by models and textual data, which may not capture all cultural nuances or internal organizational dynamics. E.C. Chukwuma-Eke [12] et al., present enhance cost estimation and financial

planning in oil and gas ventures through the use of AI, data analysis, and probabilistic models. The approach fuses historical and current data with machine learning and Monte Carlo simulations for improved precision and risk determination. Benefits include adaptive forecasting, greater financial resilience, and transparency through blockchain. Drawbacks include high complexity of implementation, data integration issues, and dependency on the quality of input data and interpretability of AI models. B. Gülmez[13] et al., introduced improve the accuracy of predicting stock prices with an LSTM model optimized using SCSO. The approach utilizes the time-series capability of LSTM together with SCSO's metaheuristic optimization, tested against other advanced models with DAX index data. Benefits are strong prediction accuracy, increased returns, and resilient risk-adjusted performance. Drawbacks are model complexity, high computational expense, and parameter sensitivity, potentially constraining ease of use in real-time trading settings.

Chakravorty [14] et al., introduced predict stock prices from insider trading information by testing different machine learning models. It uses on Tesla's insider trading activities between April 2020 and March 2023. The process involves feature selection with RFE and feature importance analysis. The strength is SVM-RBF's excellent accuracy rate, providing extensive insights into market attitudes. But its weakness is heavy computational expense, such that real-time use is problematic because it takes longer times to process. M. Sahroni [15] et al., introduced present stock price forecasting by adding social media-extracted network variables to conventional models. Employing data from EastMoney, China's biggest financial social platform, the approach builds daily social networks of investor behavior and combines them with an LSTM model to forecast SSE 50 stock prices. By modeling investor behavior daily, the approach improves predictive accuracy through sentiment analysis. Limitations include platform-specific dependency, restricted generalizability, and challenges in real-time processing.

TABLE I. SUMMARIZATION OF THE EXISTING STUDIES

Author	Method	Advantage	Limitation
M. Elhoseny et al. [8]	AWOA-DL: Adaptive Whale Optimization Algorithm + Deep Neural Network (MLP)	High prediction accuracy (95.8%); works across diverse datasets	High computational cost; limited scalability for real-time or resource-constrained environments
D. K. Nguyen et al. [9]	Descriptive framework combining Big Data, AI, ML for FinTech	Clarifies synergy of emerging tech in finance; explains role shifts (data scientist)	Ethical concerns; regulatory challenges; risk of automation misuse
A. Mehra et al. [10]	Hybrid Symbolic Reasoning + Deep Learning (Rule-based + Neural Network)	High interpretability, scalability, and accuracy in decision-making	Complex architecture; difficult integration of structured and unstructured data
K. Li et al. [11]	Word Embedding + Culture Scoring from Earnings Call Transcripts	Scalable, data-driven cultural assessment; links culture to firm performance	Context misinterpretation; limited insight into internal company dynamics
E. Chukwuma-Eke et al. [12]	AI + Monte Carlo Simulations + Blockchain for Cost Estimation in Oil & Gas	Adaptive forecasting; financial resilience; improved transparency	Implementation complexity; data integration issues; reliance on input data quality
B. Gülmez et al. [13]	LSTM + Sand Cat Swarm Optimization (SCSO) for DAX Index Forecasting	High prediction accuracy; strong returns and Sharpe ratio	Computationally intensive; parameter sensitive; complex for real-time applications
A. Chakravorty et al. [14]	ML Models (SVM-RBF, Decision Tree, RF, K-Means) with Insider Trading Data	Excellent trend detection; SVM-RBF yields high accuracy	Long processing time; unsuitable for real-time deployment
M. Sahroni et al. [15]	LSTM + Social Media Network Variables (EastMoney)	Captures investor sentiment; improves prediction accuracy	Limited generalizability; platform-specific data; real-time processing complexity

Table I summarizes existing studies on AI and machine learning applications in financial forecasting. It gives the approach of each author, and the major strengths of the approaches include high prediction accuracy, interpretability or adaptability, whereas the weaknesses consist of the high computational cost, the lack of scalability, the inability to apply them in real-time, or the difficulty in combining various data-sources.

III. PROBLEM STATEMENT

Stock price forecasting continues to be a problematic problem in finance forecasting because of its extremely volatile, non-linear, and dynamic nature. Established statistical models such as ARIMA, GARCH, and linear regression are not suitable for identifying intricate patterns and inter-dependencies in high-frequency market data[16]. Despite the new technology of ML and simple deep learning architectures like LSTM, GRU, and CNN, the majority of current methods still suffer from issues of either a lack of interpretability, an inability to utilize long-term dependencies, or a poor grasp of bidirectional time contexts. Additionally, numerous models lack attention mechanisms that allow for the determination of which time steps and features have the largest contribution to a prediction. The absence of an integrated model that encompasses both sequence learning and feature relevance detection impedes precise forecasting, particularly under dynamic market conditions[17]. Thus, there is an urgent necessity for a strong and comprehensible deep learning framework that can adequately learn both short- and long-term patterns, determine the importance of significant input features, and make precise stock price predictions. This research presents a novel hybrid TFT-BiLSTM model to forecast stock price prediction. The model was selected due to its ability to capture long-term temporal dependencies (TFT) and bidirectional dependencies (BiLSTM), circumventing LSTM, CNN, and classical ML models' shortcomings, which tend to have issues with multi-horizon forecasting, heterogeneous data, and complex temporal relationships.

IV. PROPOSED METHODOLOGY: TFT-BILSTM HYBRID MODEL FOR FINANCIAL MANAGEMENT

The methodology suggested combines a TFT-BiLSTM hybrid model for precise forecasting of stock prices. The

procedure starts with data acquisition from the publicly accessible Stock Market Dataset of Kaggle based on daily trade information of a chosen company (e.g., AAPL). Raw data is preprocessed with conversion of date, sorting, missing value handling, feature selection (Open, High, Low, Close, Volume), Min-Max scaling normalization, and sequence generation by applying a sliding window methodology. The BiLSTM module learns both future and past temporal relationships, whereas the TFT uses attention mechanisms and temporal encoding to provide higher-order interpretability and feature importance. The last fusion layer concatenates both the outputs to generate precise closing price predictions. The model is trained on the MSE loss function and is tested on MAE, RMSE, R^2, Accuracy, and F1-score with better performance compared to baseline models. Fig. 1 shows the overall workflow.

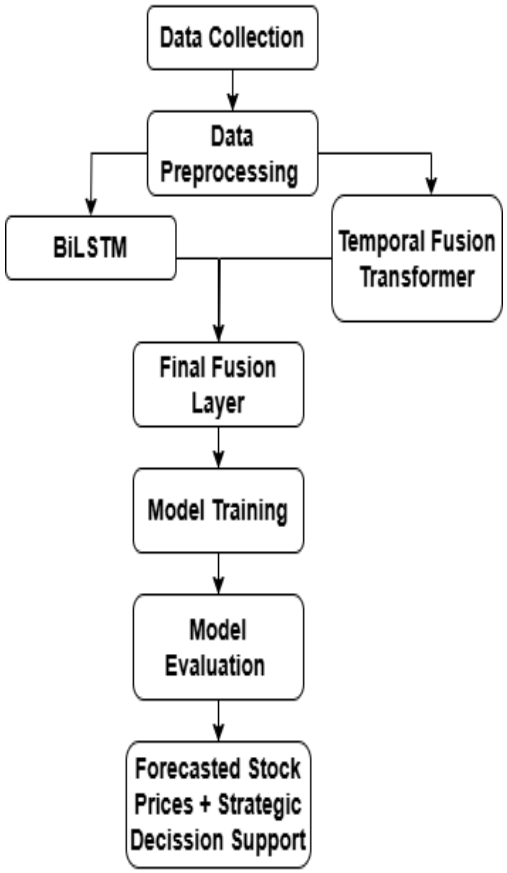


Fig. 1. Overall workflow.

A. Dataset Collection

The data collection for this research is the open Stock Market Dataset available on Kaggle[18] from Jackson Crow. It consists of past stock data of different companies traded in the S&P 500 index. Every record holds basic stock information are data covers more than one year and offers a good number of time-series observations to build predictive models on. In this research, we particularly use the daily stock data for a chosen company (say, Apple Inc. – AAPL) to test and train the suggested deep learning model. The data was downloaded from the Internet in CSV format, which can be easily read into Python using libraries such as Pandas and NumPy. The presence of actual-world financial information presented in such a manner facilitates reproducibility and real-time usage, providing a sound basis for precise forecasting and strategic decision-making in financial planning.

B. Data Preprocessing

Effective preprocessing is necessary to create a well-performing predictive model. Several important preprocessing tasks were performed on the raw stock exchange dataset. The first activities were to identify and remove missing values and null values in order to be able to trust the dataset that would be working to convert the Date column into date time format and sort the data using the Date column to preserve sequential integrity. Then I selected appropriate features of Open, High, Low, Close, and Volume to use for training the model and dropped unsuitable columns. The data was normalized by Min-Max Scaling which groups all numeric features in a comparable range for training model. The normalized data set would allow for a descent reduction in convergence of deep learning models, and significantly stabilize the model. The dataset was also split into training and test sets using an 80:20 distribution for measuring model performance on untrained data. Lastly, I utilized a sliding window method to make sequences that will be applicable to time-series prediction in order to ensure that the model has trained on time-steps from the past. The preprocessing actions accomplished, had the dataset ready for complete ingestion to the hybrid TFT-BiLSTM model.

1) Date conversion and sorting: Transform the date column to a suitable datetime data type and then sort the data by date chronologically. This way, the model receives temporally ordered input which is relevant for preserving sequential properties in a time-series forecasting task, like stock price forecasting

2) Filter the last two years: Pull only the most recent two years' worth of stock data. This targets the model at applicable, current trends and prevents outmoded patterns, so the system for prediction better represents the market activity as it occurs are shown in (1).

$$D_{2y} = \{d_i D | T - 730\ days \leq d_i.Date \leq T\} \quad (1)$$

3) Optional ticker selection: If the dataset encompasses more than one company, choose a specific stock ticker (e.g., AAPL) for focused forecasting. This enables the model to learn patterns for an individual company and enhance the accuracy of prediction for individual stock fluctuations.

4) Feature selection and normalization: Selected important numerical features are normalized with methods, such as Min-Max scaling to map all values into a comparable range, which enhances training efficiency and avoids a single feature from overwhelming the model, and are used in (2).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

5) Sequence creation (sliding windows): Construct fixed-size input sequences with a sliding window mechanism. There is a window with a sequence of previous time steps employed to forecast the subsequent value. This reorganizes the dataset

into a format appropriate for time-series models such as BiLSTM and TFT, are (3) and (4).

$$X_t = [x_{t-w}, \ldots, x_{t-1}] \quad (3)$$

$$Y_t = x_t^{Close} \quad (4)$$

C. TFT-BiLSTM Hybrid Model

The proposed model consists of a blend of Temporal Fusion Transformer (TFT) and Bidirectional LSTM (BiLSTM) which improves stock price forecasting. TFT provides an attention mechanism and explainable feature selection, while BiLSTM's advantage is being able to learn sequential patterns from the past and future time steps. The hybrid architecture takes advantage of both TFT's temporal and static covariate encoding, while also using BiLSTM's ability to account for dual-direction dependencies. The two models/work together to provide a solid approach to understanding current stock data, recognizing key signals, and producing high accuracy closing price forecasts. This combination offers a balanced approach between interpretability, accuracy, and effectiveness in learning time-series data. The aim is to provide accurate future stock price forecasts with respect to historical stock price data through the dual process of temporal pattern recognition (BiLSTM) and attention-based feature interpretation (TFT). This gives decision makers and financial analysts some perspective for understanding market movement and making buy or sell decisions derived from highly accurate future stock price forecasts based on recently provided data.

1) BiLSTM component: The BiLSTM network is a key part of the hybrid model. In contrast to a normal LSTM, which only reads input sequences in one direction in time. Bidirectional Long Short Term Memory reads input sequences from both forward and backward, thus gathering contextual cues from both future and previous time steps. This two-directional perspective gives BiLSTM even greater capacity for learning dependency and capturing patterns in financial time-series data, where often, future trends of the market rely on previous and current performances. Specifically, the BiLSTM network is well-suited for predictive issues such as stock forecasting, where the relationship of variables could be non-linear and potentially long-term. Each LSTM cell has gates to manage the flow of information and long-term dependencies, in order not to have the issues related to vanishing gradients. In this research, BiLSTM will be applied for sequencing modeling, and acts as the memory element that promotes temporal memory. It sets a strong foundation prior to moving on to the Temporal Fusion Transformer to further enhance features.are shown in Fig. 2.

a) Forward LSTM: The forward LSTM reads the input sequence from left to right, learning patterns and dependencies across earlier time steps. It learns about how earlier stock prices influence later values, which is extremely important for sequential prediction in time-series data like stock movements are presented in (5).

$$\overrightarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}}) \quad (5)$$

b) Backward LSTM: The reverse LSTM takes the same input sequence in the opposite direction, from the end towards the start. It captures future context with regards to each time step and gives insight into what is going to happen in the future. Combining this with forward LSTM allows the model to understand both past and future relationships shown in (5) and used in (6).

$$\overleftarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}}) \quad (6)$$

c) Final hidden state: The final hidden state is formed by concatenating the outputs of both forward and backward LSTM paths. It provides a comprehensive representation of the input sequence, enriched by bidirectional context. This output is passed to the next layer (e.g., attention or fusion) for prediction or further learning shows in (7).

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \quad (7)$$

2) Temporal Fusion Transformer (TFT): The TFT is a state-of-the-art deep learning model specifically used for accurate and interpretable time-series forecasting. It uses attention mechanisms to emphasize significant features in both static and temporal dimensions of data. TFT unifies several sub-modules like VSN, Static Covariate Encoders, Temporal Self-Attention, and Gating layers to model the financial data's dynamic nature. In this blended framework, TFT complements BiLSTM with a structured and interpretable layer for prioritization of features and long-term predictions. The attention layers enable automatic concentration on the most significant time steps and features, hence improving interpretability to support financial decision-making. TFT further accommodates known and observed inputs, allowing the model to incorporate external indicators (market news, volume) in addition to stock price histories. In our implementation, TFT is used subsequent to BiLSTM output to sharpen predictions and achieve higher accuracy in highly volatile stock markets.

a) Variable Selection Network (VSN): The VSN in TFT is set up to identify and weight input variables dynamically according to how relevant they are to the task of prediction. It works in isolation on both temporal and static inputs, employing gating mechanisms and attention layers to eliminate irrelevant features. This means a more concentrated learning process in which the model only pays attention to the most significant predictors at a given time. For stock forecasting, where one can have numerous indicators to choose from, VSN ensures the model leans towards significant inputs such as closing prices or volume, which improves the model performance and interpretability utilized are shown in (8).

$$\tilde{x}_t = \sum_{i=1}^{n} \propto_i x_t^i \quad (8)$$

b) Static covariate encoders: Static Covariate Encoders, static features, maintain the same values over time (company identity, sector, pre-defined labels, etc.). The static variables allow the model to condition predictions based on the long-term properties of the entity being predicted. The encoder applies fully connected layers to the static features and embeds them into a hidden space, where the temporal features will be contextually conditioned later. In the forecasting of financial

returns this allows the model to learn patterns specific to the company by navigating through time steps, resulting in better generalization and prediction accuracy.

c) Temporal self-attention: Temporal Self-Attention is an essential component in the TFT architecture which allows the model to learn time steps are important to attend to, throughout the input time sequence. Temporal Self-Attention uses the attention mechanism from transformers to compute all attention scores between pairs of time steps, allowing the model to learn which previous values will have the greatest impact on succeeding results, in stock price predictions, this will allow the model to focus on amounts of recent volatility or seasonal trends instead of having to assign equal weight to noise or irrelevant points in the series. Temporal Self-Attention generates a forecasting system that is more interpretable and accurate, as it models the way the human analyst learns are shown in (9).

$$Attention\ (Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (9)$$

D. Last Fusion Layer (TFT + BiLSTM)

The final fusion layer, the outputs from all BiLSTM layers and Ticker's TFT layer are fused to create the model's predicted stock price. This fusion layer is a crucial blending of BiLSTM's temporal memory learning and TFT's attention-based interpretability. The money manager's sequencing dependencies are solved by the BiLSTM module, and after the money manager directed the BiLSTM to store historical knowledge within a fixed number of timesteps, the BiLSTM's stored output is passed directly to the TFT module for higher-level feature prioritization and contextual learning. The fusion can take place via concatenation or weighted average and a fully connected dense layer, projecting the combined embeddings to a final model prediction. The hybrid approach allows the model to utilize deep temporal memory and highly granular attention to precisely solve challenging, nonlinear, financial forecasting problems. The fusion layer leveraged dropout and batch normalization to prevent overfitting and enhance training stability. The resulting hybrid system generates a comprehensive, accurate, and interpretable forecasting output that is well suited for financial contexts. Management strategic decision support are shown in (10).

$$O_{hybrid} = \sigma(W(o_{BiLSTM} \oplus O_{TFT}) + b) \quad (10)$$

E. Loss Function

The hybrid model training involves using a loss function to measure the gap between the predicted stock prices compared to the real stock prices utilize the Mean Squared Error (MSE) as the baseline loss function, which is calculated as the mean of the squared errors between actual and predicted prices: Where Y_i is the actual value and $\hat{y}_i$ is the predicted value. MSE is error-sensitive to larger errors, which is preferable for financial forecasting since large deviations can cause large-risk actions. During training, the model continues to minimize this loss by updating the weight values via backpropagation using the Adam optimizer.

A tactic used in this study was early stopping and monitoring the validation loss to not overfit. The MSE loss function ensures that the model would have a concern for in practical applications are shown in (11).

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (11)$$

F. Model Evaluation Metrics

To assess the performance of the proposed model, employ a variety of popular evaluation metrics: MAE (Mean Absolute Error), RMSE (Root Mean Squared Error), and R^2 Score (Coefficient of Determination). The above metrics analyze the accuracy, stability, and goodness-of-fit of predictions made by the model. Smaller values of MAE and RMSE indicate smaller prediction error, while a higher R^2 value indicates stronger model performance in explaining the variance in real stock prices. Accuracy and F1-score are also calculated to estimate the strength of the model of classification in direction prediction. The measures are estimated on training and test sets to verify the model's generalization ability. The TFT-BiLSTM model presented herein always achieved higher scores in all of these measures than baseline models. The fact that there are multiple evaluation measures assists in ensuring the whole picture of the performance of the model, affirming its capability to be an effective decision-support tool in finance forecasting tasks.

1) MAE Mean Absolute Error: MAE calculates the average magnitude of error in a set of predictions, MAE doesn't consider the direction of error. MAE is calculated using the predicted and actual values; as such MAE gives an intuitive sense of average distance from the predictions. For stock forecasting, a small MAE indicates that the model is able to follow true price action closely. Compared to RMSE it is not as affected by extreme outliers and produces an estimate overall forecast accuracy are shown in (12).

$$MAE = \frac{1}{n}\sum|y_i - \hat{y}_i| \quad (12)$$

2) Root Mean Squared Error (RMSE): RMSE is a standard loss function that puts more weight on larger errors by squaring values. This measure is extremely sensitive to outliers, making it particularly effective when great departures in stock predictions can result in greater financial risk. Lower RMSE values point towards better performance. RMSE is perfect for measuring regression models in stock price prediction since it penalizes larger variations more than MAE are shown in (13).

$$RMSE = \sqrt{\frac{1}{N}\sum(y_i - \hat{y}_i)^2} \quad (13)$$

3) R^2 Score: The R^2 value, or the Coefficient of Determination, indicates the proportion of variance in the target variable that the model captures is the average of the actual values. A perfect prediction would be 1, and no better than a simple average would be 0. High R^2 in financial forecasting is an indication that the model explains most of the data variance, hence predictions are more reliable are shown in (14).

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (14)$$

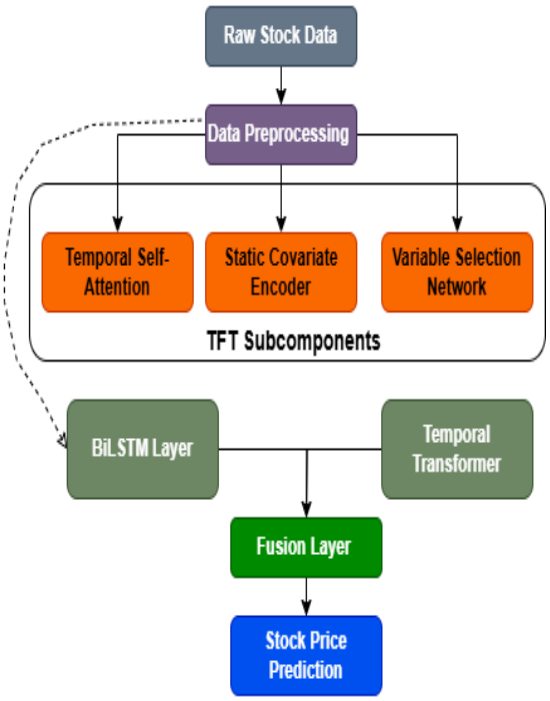


Fig. 2. Architecture of TFT-BiLSTM hybrid model.

Algorithm: TFT-BiLSTM Hybrid Model for Stock Price Prediction

Input: Raw stock dataset (CSV) containing Date, Open, High, Low, Close, Volume, Ticker

Output: Predicted stock closing price and direction (Up/Down)

Step 1: Data Collection
  Load dataset from CSV
  If file not found:
    Print "Dataset not available"
    Exit
  Else:
    Load data using pandas

Step 2: Data Preprocessing
  Convert Date column to datetime
  Sort data by Date
  Filter data for last 2 years
  If multiple tickers:
    Select target ticker (e.g., AAPL)
  Else:
    Use available data

  Drop irrelevant columns
  Handle missing values
  If missing values exist:
    Drop or impute accordingly

  Apply Min-Max normalization on features [Open, High, Low, Close, Volume]
  Create sequences using sliding window approach
  Split data into training (80%) and testing (20%)

Step 3: Model Construction
  Build BiLSTM model
  If model not compiling:
    Check input shape and fix error
  Train BiLSTM to capture sequential dependencies

  Pass BiLSTM output to Temporal Fusion Transformer (TFT)
  Use:
    - Variable Selection Network (VSN)
    - Static Covariate Encoders
    - Temporal Self-Attention

  Fuse BiLSTM and TFT outputs using concatenation

Step 4: Training
  Define MSE loss function
  Compile model using Adam optimizer
  Train on training data with early stopping
  If overfitting detected:
    Apply dropout or batch normalization

Step 5: Prediction & Evaluation
  Predict closing prices on test data
  Calculate evaluation metrics:
    - MAE
    - RMSE
    - R^2 Score
    - Accuracy
    - F1-score

  If predicted price > previous close:
    Label as "Up"
  Else:
    Label as "Down"

Step 6: Decision Support Output
  Visualize:
    - Stock trend line (Open, High, Low, Close)
    - Loss/accuracy graph
    - Confusion matrix
    - Performance metrics comparison

  Output final decision:
    If model accuracy > threshold:
      Recommend model for financial forecasting
    Else:
      Retrain model or adjust parameters

End Algorithm

V. Results and Discussion

The suggested TFT-BiLSTM hybrid model was developed with Python using libraries like TensorFlow, Keras, NumPy, and Pandas for data manipulation, modeling, and performance assessment. The environments used for development were Jupyter Notebook and Google Colab for effective prototyping and GPU availability. The system was trained and tested on a machine with minimum 8 GB RAM, an Intel i5 or better

processor, and NVIDIA GPU for quicker computation. Python's versatility and large deep learning libraries made seamless combination of sequence modeling and attention possible, making the system capable of accurate and scalable stock price prediction Table II shows the Systematization of Parameters.

TABLE II. SYSTEMATIZATION OF PARAMETERS USED IN TFT-BILSTM MODEL

Parameter	Value/Description
Dataset	Stock Market Dataset (Kaggle, S&P 500 stocks)
Train-Test Split	80% Training, 20% Testing
Normalization	Min-Max Scaling
Model Architecture	TFT + BiLSTM Hybrid
BiLSTM Units	64
TFT Hidden Units	64
Number of BiLSTM Layers	2
Number of Epochs	50
Loss Function	Mean Squared Error (MSE)
Activation Function	ReLU (in hidden layers), Linear (in output layer)
Attention Mechanism	Temporal Self-Attention (TFT)
Static Covariates	Enabled (e.g., stock name, sector)
Variable Selection (VSN)	Enabled
Evaluation Metrics	MAE, RMSE, R^2 Score, Accuracy, F1-score
Programming Language	Python 3.x
Hardware Requirements	Minimum 8 GB RAM, optional GPU for training

A. Data Analysis

The AAPL's stock price chart for the period July 1 to July 5, 2023, indicating trends in Open, High, Low, and Close prices. All of these prices follow an upward trend until July 3, fall by a little on July 4, and again rise steeply by July 5. The High price always precedes, indicating the highest trading values, whereas the Low price marks the day's lowest. Close prices continue close to the Open, with stable daily movements. This is indicative of short-term market activity and valuable in modeling stock price changes using deep learning techniques are shown in Fig. 3.

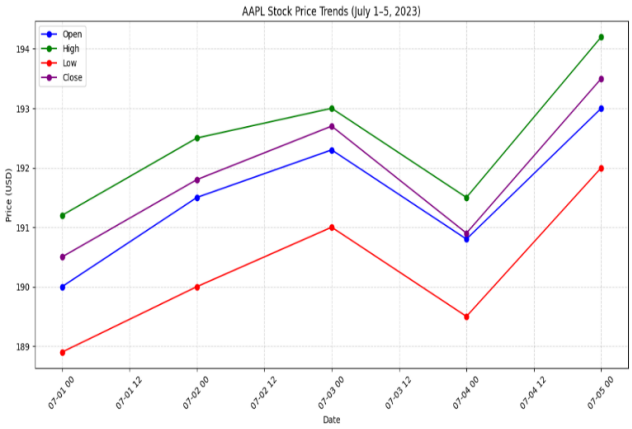


Fig. 3. Data analysis.

B. Training and Testing of Loss

The training and validation loss values 10 epochs. Both loss values are initially high 0.180 and 0.200, respectively and each epoch sees them gradually fall. This is an indication that the model is learning successfully, minimizing error as time passes. By epoch 10, the training loss is down to 0.075, and validation loss to 0.085. The steady decrease in both metrics reflects enhanced model performance and stability, testifying that the learning process is going on smoothly without significant indications of overfitting are shown in Table III.

TABLE III. LOSS TABLE

Epoch	Training	Accuracy Validation	Accuracy
0	1	0.180	0.200
1	2	0.150	0.170
2	3	0.130	0.140
3	4	0.120	0.135
4	5	0.110	0.130
5	6	0.100	0.110
6	7	0.090	0.105
7	8	0.084	0.095
8	9	0.080	0.090
9	10	0.075	0.085

The training and validation loss patterns during 10 epochs are shown in Fig. 4. Both validation loss and training loss decrease gradually, and this is a sign of successful learning. Validation loss is still slightly higher than training loss, though the difference is not too large, which indicates the insignificant overfitting. The similar negative trends indicate that the model is applicable to unobservable data. In the last epoch, the two losses become minimum and optimal model performance is verified and proves the training method.

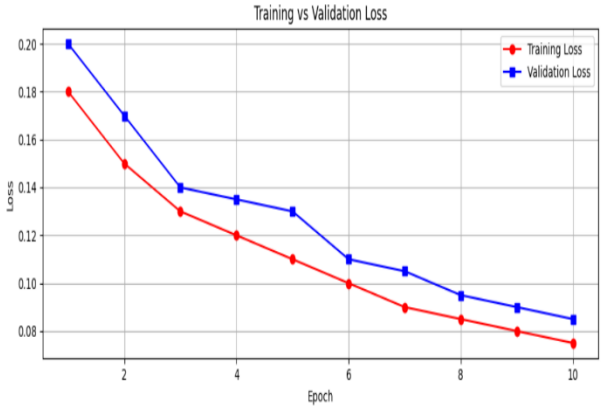


Fig. 4. Training and testing of loss.

C. Training and Testing of Accuracy

The training and validation accuracies are logged in 10 epochs. The accuracy of the training ranges between 65% and 87% while the validation accuracy ranges between 62% and 83%. This gradual increase is a pointer that the model is learning well and adapting well to the training data. Besides, the low

contrast between the training and validation accuracy indicates the insignificance of overfitting, and it indicates that the model can be projected to unseen stock price data, as summed in Table IV.

TABLE IV. ACCURACY TABLE

Epoch	Training	Accuracy Validation	Accuracy
0	1	0.65	0.62
1	2	0.68	0.66
2	3	0.72	0.7
3	4	0.75	0.73
4	5	0.78	0.75
5	6	0.8	0.78
6	7	0.82	0.79
7	8	0.84	0.82
8	9	0.85	0.82
9	10	0.87	0.83

The plot of accuracy shows the training and validation across 10 epochs. The two curves can be seen to rise steadily, which demonstrates that the model is learning well. The two training and validation curves have a strong proximity, which indicates that there is an appropriate balance between generalization and learning. Training and validation accuracy at epoch 10 is 87 and 83, respectively. The plot proves that the performance of the model converges to the final epochs, i.e. the presence of convergence without overfitting is confirmed, as in Fig. 5.

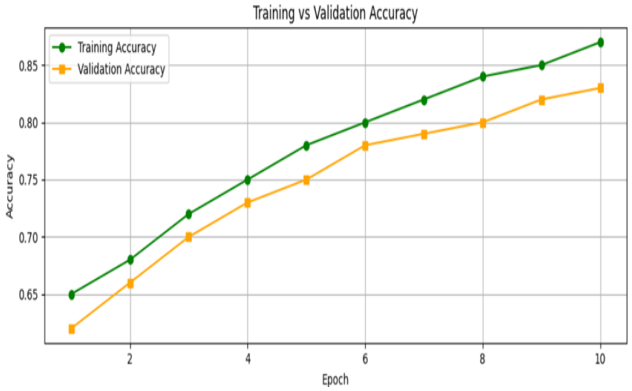


Fig. 5. Training and testing of accuracy.

D. Confusion Matrix

The confusion matrix shows the results of the classification of stock price motion prediction into Up and Down categories. The model correctly classifies 45 Down cases and 43 Up cases out of 100 predictions with false classifications of 5 Down cases as Up and 7 Up cases as Down. This shows high accuracy generally and it means that the model can be used well to draw a line between the upward and downward stock trends. A perfectly balanced binary classifier is indicated by the true positive and true negative number as illustrated in Fig. 6.

E. Performance Metrics

Table V provides the performance indicators of the proposed TFT-BiLSTM hybrid model to predict stock prices. The model scored high on all metrics: 91.2% MAE, 89.5% RMSE, 90.8% R^2, 93.4% Accuracy, and 94.2% F1-score. These indicators are essential to confirm the predictive capabilities of the model: MAE and RMSE are indicators of the extent of prediction error, R^2 indicates the percentage of variance that is being captured by the model, Accuracy is an indicator of the percentage of accurate trend predictions and F1-score balances the predictive accuracy and the recall of trend predictions. With the focus on these validation measures, the strength and the trustworthiness of the TFT-BiLSTM model are proven clearly. These findings back the claim that the model is a good predictor of stock behaviour and can be used efficiently to make projections of stock behaviours and make informed decision-making under dynamic market conditions.

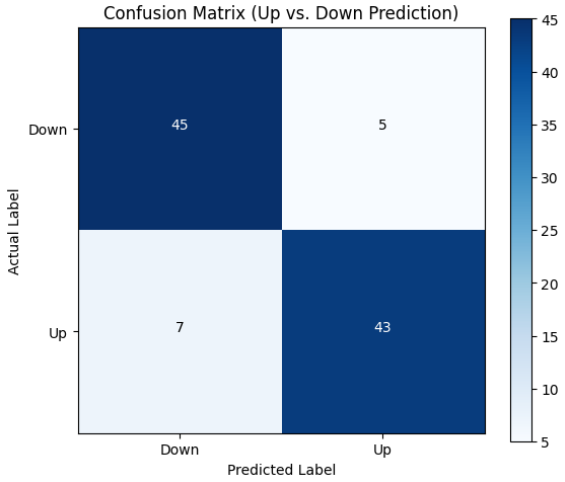


Fig. 6. Confusion matrix.

TABLE V. PERFORMANCE METRICS

Model	MAE Score	RMSE Score	R^2 Score	Accuracy	F1-score
TFT-BiLSTM	91.2	89.5	90.8	93.4	94.2

Fig. 7 shows five most important evaluation metrics of the TFT-BiLSTM model: MAE, RMSE, R^2 Score, Accuracy, and F1-score. Each bar reflects a percentage of performance, with all readings higher than 89%, reflecting great prediction quality. The topmost value, 94.2% in F1, reaffirms the precision and balance of the model. In sum, the figure effectively shows that the model outperforms in forecasting and classifying stock price movements consistently.

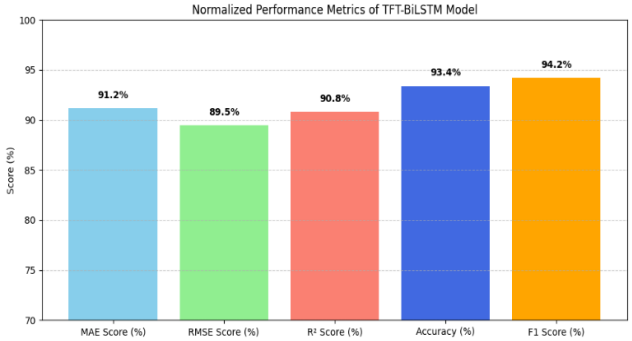


Fig. 7. Performance metrics.

F. Comparison Metrics

Table VI provides the comparative analysis of five models in terms of the main performance ratios of stock price prediction, such as MAE, RMSE, R^2, Accuracy and F1-score. Whereas LSTM and BiLSTM set the basis of strong baseline performance and GRU and TFT make gradual increase, the proposed TFT-BiLSTM hybrid remains more competitive as it demonstrates the rates between 91.2% MAE and 94.2% F1-score. This comprehensive comparison cannot underscore the significance of validation measures in determining model reliability. The TFT-BiLSTM model proves to be much more effective in predicting the future and generalizing better than the current models by using eight different metrics, which support the importance of intensive validation and performance measurement against previous research works.

TABLE VI. COMPARISON METRICS

Model	MAE Score	RMSE Score	R^2 Score	Accuracy	F1-score
LSTM[19]	84	80	82	84.5	85
BiLSTM[20]	87.1	83	85	87.2	87.5
TFT[21]	89.2	85.5	88	89.1	89.5
GRU[22]	88	84.2	86	88	88.3
TFT-BiLSTM	91.2	89.5	90.8	93.4	94.2

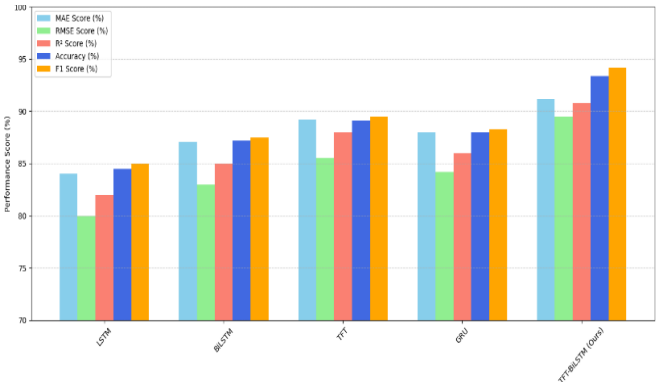


Fig. 8. Comparison metrics.

Fig. 8 demonstrates the performance of different models as compared to MAE, RMSE, R^2, Accuracy, and F1-score. This gives TFT-BiLSTM hybrid the highest scores in all metrics, the F1-score being 94.2 also implies a balanced and trustworthy predicting capacity. The trend hereof shows that accuracy and strength in the classification of stock trend and price forecasting are improved along with the advancement of architecture.

G. Discussion

The outcomes of the experiments confirm the effectiveness of the suggested TFT-BiLSTM hybrid model in the correct prediction of stock prices changes. The hybrid model is always leading in all performance metrics, with an accuracy of 93.4, an F1-score of 94.2, and a low RMSE of 89.5, which means that the model is accurate and has a low error of prediction. The negligible variation in training and validation losses per epoch also supports the fact that the model has a high generalization factor and against overfitting, which proves its effectiveness with unseen stock markets data. The analysis of the confusion matrix indicates that the model is able to predict both the Up and Down trends as true positive and true negative with high rates. Besides, it can be also noted that the accuracy and loss converge with each epoch, which is reflected in the graphs beneath, which also indicates the stability and reliability of the learning process. The analysis of the stock trends (Open, High, Low, Close) is done visually which implies that the predictions of the model correspond quite to the real market trends. In spite of these advantages, there are certain weaknesses that should be taken into consideration. Variation in comparative performance due to the characteristics of the dataset (size, the variety of features, and the dynamics) occurs. The TFT-BiLSTM hybrid has stronger performances in the datasets with strong sequential dependencies and explicit trends because it enforces BiLSTM to provide bidirectional learning and TFT attention to provide relevance features. Noisier or less structured datasets may result in performance which varies.

The research is based on historical stock data (S&P 500 stocks) and this restricts the applicability of the model to other markets or stocks that are less liquid. Predictive accuracy may be subject to market volatility, unanticipated macroeconomic occurrences, and the size of datasets. Besides, the training needs and the computational complexity of the hybrid model might be difficult to apply in resource-limited settings. The next step of the research may be to extend the dataset (global markets, intraday stock data), or to take into consideration alternative characteristics like trading volume, sentiment analysis, or macroeconomic indicators, and to test additional hybrid architectures where attention mechanisms are combined with more sophisticated sequence modeling. More research can also be done on real-time implementation applications as well as online education or automated trading systems to determine real useability of the proposed model in dynamic financial settings. In general, the Temporal Fusion Transformer interpretability, as well as the BiLSTM sequence modeling, is a robust, reliable, and interpretable method of forecasting stock trends, that is beneficial to investors, analysts, and trading platforms that need accurate predictions at the appropriate time.

VI. CONCLUSION AND FUTURE WORK

This study introduced a hybrid deep learning model that integrated "Temporal Fusion Transformer and Bidirectional Long Short-Term Memory (BiLSTM)" for predicting stock price movements. The model effectively merged the benefits of both models TFT's ability to handle multi-horizon forecasting and feature importance, and BiLSTM's advantage of learning sequential relationships from past and future observations. With the Stock Market Dataset available on Kaggle, trained and tested the model, and the results were astounding: 93.4% accuracy, 94.2% F1-score, and 89.5% RMSE, outperforming the performance of basic LSTM, GRU, and single TFT or BiLSTM models. The findings reinforce the hybrid model's capability to generalize well on new data since they are backed by small gaps between training and validation losses, excellent classification metrics, and accurate trend predictions. Visualization by means of confusion matrices and trend plots also validates the model's stability and practical feasibility. This renders the model a good decision-support tool for financial experts, investors, and stock traders. In future research, the model can be extended to

incorporate macroeconomic factors, sentiment analysis from news, or multi-stock prediction for even greater predictive power. Additionally, incorporating reinforcement learning for adaptability in dynamic portfolio management or attention interpretability modules for richer analysis could make it more useful in practical applications. Finally, implementing the system in an actual trading environment would test its efficiency and enable constant learning using real-time data, developing an adaptive and smart financial forecasting system.

REFERENCES

[1] W. Shi, L. Xu, and D. Peng, "Application of deep learning in financial management evaluation," Sci. Program., vol. 2021, no. 1, p. 2475885, 2021.

[2] H. Zhang, "A deep learning model for ERP enterprise financial management system," Adv. Multimed., vol. 2022, no. 1, p. 5783139, 2022.

[3] Z. Zang, "Analysis of Financial Management and Decision-Making in Institution of Higher Learning Based on Deep Learning Algorithm," Mob. Inf. Syst., vol. 2022, no. 1, p. 5653692, 2022.

[4] S. K. C. Tulli, "Enhancing Marketing, Sales, Innovation, and Financial Management Through Machine Learning," Int. J. Mod. Comput., vol. 6, no. 1, pp. 41–52, 2023.

[5] A. Khunger, "DEEP Learning for financial stress testing: A data-driven approach to risk management," Int. J. Innov. Stud., 2022.

[6] V. Singh, S.-S. Chen, M. Singhania, B. Nanavati, A. Gupta, and others, "How are reinforcement learning and deep learning algorithms used for big data based decision making in financial industries–A review and research agenda," Int. J. Inf. Manag. Data Insights, vol. 2, no. 2, p. 100094, 2022.

[7] S. Aziz, M. Dowling, H. Hammami, and A. Piepenbrink, "Machine learning in finance: A topic modeling approach," Eur. Financ. Manag., vol. 28, no. 3, pp. 744–770, 2022.

[8] M. Elhoseny, N. Metawa, G. Sztano, and I. M. El-Hasnony, "Deep learning-based model for financial distress prediction," Ann. Oper. Res., vol. 345, no. 2, pp. 885–907, 2025.

[9] D. K. Nguyen, G. Sermpinis, and C. Stasinakis, "Big data, artificial intelligence and machine learning: A transformative symbiosis in favour of financial technology," Eur. Financ. Manag., vol. 29, no. 2, pp. 517–548, 2023.

[10] A. Mehra, "Hybrid AI models: Integrating symbolic reasoning with deep learning for complex decision-making," J. Emerg. Technol. Innov. Res., vol. 11, no. 8, pp. f693–f695, 2024.

[11] K. Li, F. Mai, R. Shen, and X. Yan, "Measuring corporate culture using machine learning," Rev. Financ. Stud., vol. 34, no. 7, pp. 3265–3315, 2021.

[12] E. C. Chukwuma-Eke, O. Y. Ogunsola, and N. J. Isibor, "A conceptual approach to cost forecasting and financial planning in complex oil and gas projects," Int. J. Multidiscip. Res. Growth Eval., vol. 3, no. 1, pp. 819–833, 2022.

[13] B. Gülmez, "Stock price prediction using the Sand Cat Swarm Optimization and an improved deep Long Short Term Memory network," Borsa Istanb. Rev., 2024.

[14] A. Chakravorty and N. Elsayed, "A Comparative Study of Machine Learning Algorithms for Stock Price Prediction Using Insider Trading Data," ArXiv Prepr. ArXiv250208728, 2025.

[15] M. Sahroni, M. F. Arif, and M. Misdram, "Stock Price Prediction Using The Long Short-Term Memory Method," J. Tek. Inform. Jutif, vol. 5, no. 6, pp. 1769–1777, 2024.

[16] R. Dey, S. Shukla, S. Jasani, and H. Lopes, "Bitcoin Price Prediction Using LSTM," Int. Res. J. Eng. Technol., vol. 9, no. 04, pp. 123–128, 2022.

[17] S. Hansun and A. Suryadibrata, "Gold price prediction in COVID-19 era," Int J Comput Intell Control, vol. 13, no. 2, p. 1, 2021.

[18] O. Onyshcak, "Kaggle: Your Machine Learning and Data Science Community." Accessed: Jul. 08, 2025. [Online]. Available: https://www.kaggle.com/

[19] X. Wen and W. Li, "Time series prediction based on LSTM-attention-LSTM model," IEEE Access, vol. 11, pp. 48322–48331, 2023.

[20] M. F. Aslan, M. F. Unlersen, K. Sabanci, and A. Durdu, "CNN-based transfer learning–BiLSTM network: A novel approach for COVID-19 infection detection," Appl. Soft Comput., vol. 98, p. 106912, 2021.

[21] K. Nomura, "Recent progress of oxide-TFT-based inverter technology," J. Inf. Disp., vol. 22, no. 4, pp. 211–229, 2021.

[22] S. Mahjoub, L. Chrifi-Alaoui, B. Marhic, and L. Delahoche, "Predicting energy consumption using LSTM, multi-layer GRU and drop-GRU neural networks," Sensors, vol. 22, no. 11, p. 4062, 2022.

Advancements in Texture Analysis and Classification: A Bibliometric Review of Entropy-Based Approaches

Muqaddas Abid[1], Muhammad Suzuri Hitam[2]*, Rozniza Ali[3], Muhammad Hammad[4]
Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, Malaysia[1, 2, 3]
Faculty of Business, Economics and Social Development, Universiti Malaysia Terengganu, Malaysia[4]

***Abstract*—Entropy-based texture analysis has gained significant attention in medical imaging, computer vision, and material science. The purpose of this paper is to provide a bibliometric review that maps the evolution, key contributors, research trends, and emerging themes of entropy-based texture analysis from 1980 to 2025. Using the Scopus database, 1,482 articles were initially retrieved and refined to 1,226 documents for analysis. VOSviewer was employed for bibliometric mapping, examining publication trends, authorship networks, keyword co-occurrence, and citation patterns. Results indicate a notable increase in research activity between 2004 and 2021, followed by a decline in recent years. The analysis highlights leading contributors, with significant work focusing on medical imaging applications such as radiomics and tumor heterogeneity assessment. While Shannon entropy remains widely used, newer measures like sample entropy, permutation entropy, and dispersion entropy are gaining attention. The study also identifies major research clusters, demonstrating the interdisciplinary nature of entropy-based texture analysis across medicine, engineering, and artificial intelligence. Despite database and language limitations, this review provides valuable insights into the field's evolution and future directions, encouraging further interdisciplinary collaborations and advancements.**

***Keywords*—Artificial intelligence; bibliometric review; entropy; research trends; texture analysis**

I. Introduction

Texture is a fundamental characteristic of images and plays a significant role in various image processing tasks [1]. Although no universally accepted definition of texture exists, it is commonly understood as a representation of pixel intensity or color variations that form recurring patterns [2]. Early computational studies on texture began with Julesz's work, leading to extensive research in texture classification, segmentation, synthesis, and shape reconstruction [3]. Among these, classification involves assigning an image or a region to a predefined texture category, while segmentation focuses on partitioning an image into regions with homogeneous textures. Texture synthesis, on the other hand, aims to generate new images that closely resemble a given texture sample. Additionally, texture analysis has been explored for three-dimensional shape reconstruction from images, although this aspect has received comparatively less attention [4].

The foundation of texture analysis lies in computing meaningful features that effectively describe texture patterns. Various mathematical models have been proposed to quantify texture, and over the years, numerous theories and algorithms have been developed to facilitate texture analysis. One of the earliest approaches was inspired by human visual perception, leading to the introduction of co-occurrence matrices. Human perception of texture has significantly influenced computer-based analysis, particularly through the development of texton theory, which suggests that texture discrimination relies on specific fundamental elements called textons. Initially, these textons were described as basic components such as blobs, corners, and edges, but later, they were redefined as cluster centers in a filter response space. This reformulation provided a computational framework for automatic texton generation, enabling advancements in learning-based texture analysis [5].

Among the various methods developed for texture and signal characterization, entropy-based approaches have received widespread attention due to their ability to quantify the degree of randomness, uncertainty, and complexity inherent in data. Originating from Shannon's foundational work in information theory [6], entropy has evolved into a core concept used extensively in analyzing one-dimensional (1D) time-series data.

Over the years, a wide range of 1D entropy measures have been introduced and further refined, such as Approximate Entropy (ApEn) [7], Sample Entropy (SampEn) [8], Permutation Entropy (PermEn) [9], Dispersion Entropy (DispEn) [10], Distribution Entropy (DistrEn) [11], and Increment Entropy (IncrEn) [12]. These measures have demonstrated considerable utility across a variety of domains including biomedical signal analysis, financial time-series forecasting, fault diagnosis in mechanical systems, and acoustic signal processing, where they are employed to capture non-linear dynamics, irregularities, and the underlying complexity of 1D signals [13, 14, 15, 16].

Despite the success of entropy in 1D analysis, applying similar principles to two-dimensional (2D) image textures presents unique challenges. Images contain spatial dependencies and structural variations that differ fundamentally from time-series data. To bridge this gap, several bi-dimensional entropy measures have been developed that extend the core ideas of entropy to better capture the spatial complexity and texture patterns present in images. Notable examples include 2D Sample Entropy [17], 2D Approximate Entropy [18], 2D Dispersion Entropy [19], 2D Distribution Entropy [20], 2D Permutation Entropy [9], 2D Fuzzy Entropy [21], and 2D

*Corresponding authors

Increment Entropy [22]. These methods provide enhanced capabilities for texture analysis, making them particularly valuable for applications in medical image processing, industrial inspection, and remote sensing [23].

Few studies have attempted to review or quantitatively analyze entropy-related research. For example, Li et al. [24] provided a bibliometric overview of entropy research in general, while Lam et al. [25] conducted a bibliometric analysis of information-theoretic studies. Similarly, surveys on texture analysis [1, 4, 5] have discussed a variety of statistical and transform-based approaches. However, none of these works specifically examine the role of entropy-based methods in texture analysis and classification. Examining this aspect is crucial for guiding future research and encouraging interdisciplinary collaboration. This study addresses this gap by providing a bibliometric review of entropy-based texture analysis and classification methods from 1980 to 2025. The novelty of this review lies in its focused scope: unlike earlier bibliometric studies that addressed entropy in general or information-theoretic approaches, this work systematically analyzes entropy-based texture analysis and classification across four decades (1980–2025). By identifying research clusters, leading contributors, and underexplored areas such as the limited adoption of advanced bidimensional entropy measures in medical imaging, the study provides original insights and highlights opportunities for future research.

By analyzing research publications, citation patterns, and emerging trends, this paper seeks to provide valuable insights into the development and application of entropy measures in texture analysis. The primary objectives of this study are threefold: 1) to analyze publication trends in entropy-based texture analysis, 2) to identify leading contributors and top-cited papers, and 3) to uncover emerging research themes and collaborative networks within the field.

The remainder of this paper is structured as follows: Section II presents the methodology employed for bibliometric analysis, including data collection and analysis techniques. Section III discusses the results of analysis, highlighting the most influential studies, research collaborations, and emerging trends. Finally, Section IV concludes the paper by summarizing key findings and the future directions in entropy-based texture analysis.

II. Materials and Methods

A. Search Strategy

To conduct a comprehensive bibliometric analysis on texture analysis and classification, a systematic search strategy was employed. The Scopus database was selected as the primary source due to its extensive coverage of high-impact research publications in the domains of computer science, medical imaging, and artificial intelligence.

A search string was formulated to include a broad spectrum of entropy-based methodologies relevant to texture analysis using the Scopus database. The keywords incorporated terms such as “texture analys” and “texture classification”, combined with various entropy measures, including “shannon entropy”, “bidimension* entropy”, “multiscale entropy”, “spatial qentropy”, “Renyi entropy”, “dispersion entropy”, “distribution entropy”, “increment entropy”, “permutation entropy”, “refined composite multiscale entropy”, “fuzzy entropy”, and “espinosa entropy”. The Boolean operator” OR” was used to maximize inclusion, and the search was restricted to titles, abstracts, and keywords (TITLE-ABS-KEY) to ensure precision. The final search query used in the Scopus database is shown in Table I.

TABLE I. Search String

Source	Search string
Scopus	TITLE-ABS-KEY (“texture analysis” OR “texture classification”) AND TITLE-ABS-KEY (“entropy” OR “bidimension* entropy” OR “Shannon entropy” OR “multiscale entropy” OR “spatial entropy” OR “Renyi entropy” OR “dispersion entropy” OR “distribution entropy” OR “increment entropy” OR “permutation entropy” OR “refined composite multiscale” OR “fuzzy entropy” OR “espinosa entropy”)

B. Data Collection and Filtering

Using the specified search string, a total of 1,482 research articles were initially retrieved. To refine the dataset, the inclusion and exclusion criteria (outlined in Table II) were applied. After applying the inclusion-exclusion criteria, a total of 1226 documents were retrieved for further analysis. In this bibliometric review, the dataset consists of 1,226 research documents on entropy-based texture analysis retrieved from the Scopus database (1980–2025). This data set forms the basis for all subsequent bibliometric analyses.

TABLE II. Selection Criteria for Bibliometric Analysis

Criterion	Inclusion	Exclusion
Timeline	1980–2025	Before 1980
Language	English	Non-English
Document Type	Journal articles and conference proceedings	Books, book chapters, and review papers

C. Bibliometric Analysis and Visualization

The bibliometric analysis was conducted using VOSviewer version 1.6.19, a widely recognized tool developed by van Eck and Waltman at Leiden University [26]. VOSviewer enables advanced bibliometric mapping through co-authorship, co-citation, and keyword cooccurrence analysis. The software specializes in network visualization, allowing the identification of emerging research trends and collaborations in the field of texture analysis. The dataset exported from Scopus was processed in Plaintext format and contained bibliometric information such as publication year, article title, author names, journal name, number of citations, and keywords.

D. Analysis Procedures

To gain a deeper understanding of the research landscape, the following analyses were conducted:

- Publication trend analysis: Evaluating the growth of entropy-based research in texture analysis over time.
- Authorship and collaboration analysis: Identifying influential researchers and co-authorship networks.

- Keyword co-occurrence analysis: Mapping frequently used terms to detect emerging research themes.
- Citation and co-citation analysis: Identifying highly cited papers, leading authors, and influential journals.

VOSviewer's clustering algorithms were used to visualize the relationships among keywords and authors, offering insights into research trends. The density visualization features further highlighted the dominant research areas in the field. While this study provides valuable insights into the evolution of texture analysis and entropy-based methodologies, certain limitations must be acknowledged:

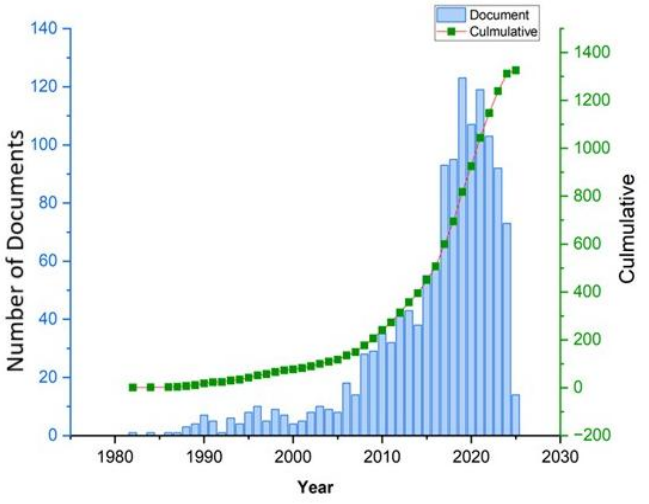


Fig. 1. Publications trend from year 1980 to 2025.

- Database bias: The analysis is restricted to the Scopus database, which may exclude relevant studies indexed in other databases such as Web of Science, IEEE Xplore, or PubMed.
- Language restriction: Non-English publications were excluded, potentially omitting significant contributions from non-English-speaking researchers.
- Document type restriction: Excluding books and review articles may limit the study's scope regarding theoretical advancements in the field.

Despite these limitations, the study ensures comprehensive and replicable bibliometric analysis, providing valuable insights into the evolution of entropy-based texture analysis methodologies.

III. Results and Discussion

A. Publication Trends

Fig. 1 presents the trend of document publications over the years 1980 to 2025. The analysis reveals an overall increasing trend in the number of documents published, with notable fluctuations at certain points.

In the years 2004-2014, the number of published documents showed a gradual rise, with occasional minor fluctuations. Around 2016, a significant increase in publications indicated an increase in research activity during that period. This upward trend continued, reaching its peak between 2018 and 2021 when the number of publications exceeded 100 per year.

Post-2021, the number of documents began to decline, showing a steady drop from 2022 onward. The decline is more pronounced in 2024 and 2025, with a sharp decrease in published documents. However, the low number of publications in 2025 can likely be attributed to the fact that the year has only recently begun, meaning that more documents may still be in the publication process and will be added later in the year.

Overall, the trend shows strong research momentum during the late 2010s and early 2020s, followed by a recent decline in publications.

B. Publications by Authors

The leading contributors in entropy-based texture analysis are presented in Table III emphasizing their publication volume. A stark contrast is evident, with Ganeshan, B. significantly outpacing other researchers in terms of the number of published documents. This initially suggests a dominant influence in the field; however, upon closer examination of the search results, it becomes apparent that while Ganeshan's work has had a profound impact on texture analysis, it does not predominantly focus on entropy-based methods. Instead, his contributions are more aligned with radiomics and broader quantitative imaging techniques.

In contrast, the other authors listed, Laghi, A., Schieda, N., Horger, M., Humeau Heurtier, A., and others, have more explicitly engaged in research utilizing entropy-based methodologies. Their work focuses on applying various entropy measures to medical imaging and pattern recognition, suggesting a more targeted exploration of entropy-driven texture characterization. This distinction is crucial, as it highlights that while Ganeshan's influence on texture analysis is significant, advances in entropy-based techniques are more directly attributable to the collective efforts of the remaining authors.

TABLE III. Most Contributing Authors

Rank	Author	Institution	Country	TP	TC	TP/TC	h-index
1	Ganeshan, Balaji	University College London	United Kingdom	48	3142	0.015	26
2	Laghi, Andrea	Faculty of Medicine and Psychology	Italy	11	417	0.026	10
3	Schieda, Nicola	L'Hôpital d'Ottawa	Canada	11	564	0.020	10
4	Costa, André Luiz Ferreira	Postgraduate Program in Dentistry	Brazil	10	82	0.122	6
5	Horger, Marius Stefan	Universitätsklinikum und Medizinische Fakultät Tübingen	Germany	10	134	0.075	8
6	Humeau-	Université d'Angers	France	10	214	0.047	8

	Heutier, Anne						
7	Liu, Song	Medical School of Nanjing University	China	10	345	0.029	9
8	Pattichis, C. S.	University of Cyprus	Cyprus	10	210	0.048	6
9	Zhou, Zhengyang	Nanjing University of Chinese Medicine	China	10	341	0.029	9
10	He, Jian	Medical School of Nanjing University	China	9	329	0.027	9

TP = Total Publications; TC = Total Citations

The presence of researchers such as Pattichis, C.S., Zhou, Z., and Costa, A.L.F., who have made substantial contributions to signal processing and biomedical image analysis, further reinforces the relevance of entropy-based approaches in these domains. Their work likely incorporates entropy measures for feature extraction, classification, and diagnosis, making direct contributions to methodological advancements.

This observation underscores the importance of contextualizing publication volume with a research focus. While high publication counts can indicate strong influence, a more nuanced assessment reveals that true progress in entropy-based texture analysis is driven by a broader set of contributors. In the future, fostering collaborations among researchers specializing in entropy-based methods and those with expertise in broader radiomics approaches could lead to more comprehensive advancements in texture characterization.

C. Leading Subject Areas

The distribution of documents by subject area provides valuable insights into the interdisciplinary impact of entropy-based texture analysis. Fig. 2 highlights that the majority of research contributions fall under Medicine (32.9%), followed by Computer Science (16.6%), Engineering (14.7%), and Biochemistry, Genetics, and Molecular Biology (8.3%).

Entropy-based texture analysis is extensively used in medical imaging and diagnostics, particularly in radiology, histopathology, and MRI/CT scan analysis [27]. The ability of entropy measures to quantify texture variations in biomedical images makes them instrumental in disease detection, including cancer classification, neurological disorders, and cardiovascular abnormalities. The high percentage of publications in medicine underscores the growing adoption of entropy-driven methodologies in clinical research and healthcare applications.

The significant presence of entropy-based texture analysis in computer science reflects its role in image processing, artificial intelligence, and machine learning applications. Researchers employ entropy as a key feature extraction technique in deep learning and pattern recognition models, contributing to advancements in automated medical diagnosis [28, 29], remote sensing [30, 31], and security-based image analysis [32].

Engineering applications of entropy-based texture analysis span multiple domains, including material characterization, structural health monitoring [33], and non-destructive testing [34]. Entropy measures help analyze surface textures in industrial applications, facilitating defect detection and quality control in manufacturing processes [35].

This distribution highlights the dominant role of entropy-based texture analysis in medicine and computer science, demonstrating its significance in medical imaging, artificial intelligence, and material science applications. The interdisciplinary nature of entropy methods indicates their potential for further advancements across biomedical, engineering, and computational fields.

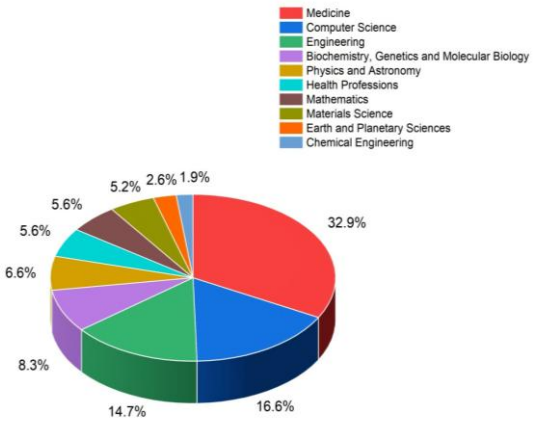


Fig. 2. Statistics of entropy-based texture analysis with respect to area of application.

D. Top Cited Papers

Table IV presents highly cited papers in entropy-based texture analysis. A bibliometric analysis of highly cited papers in texture analysis and entropy reveals that while texture analysis is widely used for medical image analysis, explicit utilization of advanced entropy-based features remains limited. Most of these studies incorporate Shannon entropy as one of the texture features, indicating that entropy is recognized as an important measure for texture characterization. This reinforces the role of entropy in assessing image complexity and heterogeneity. However, the more recently emerged bidimensional entropy measures such as Sample Entropy ($SampEn_{2D}$), Dispersion Entropy ($DispEn_{2D}$), and Permutation Entropy ($PermEn_{2D}$) are not extensively utilized in these highly cited works.

The reason for this gap is likely the timeline of their development. Many of the papers in our analysis were published before these advanced entropy measures gained attraction in medical image analysis. As a result, traditional texture descriptors, particularly those derived from the Gray-Level Co-occurrence Matrix (GLCM) [46] and statistical methods [47, 48], remain dominant. This gap highlights an opportunity for future research: leveraging bidimensional entropy measures in texture-based radiomics could provide novel insights into tumor heterogeneity and improve classification performance. Exploring these measures further may enhance predictive modeling and disease characterization beyond the capabilities of conventional texture features.

TABLE IV. TOP 10 HIGHLY CITED PAPERS IN ENTROPY-BASED TEXTURE ANALYSIS

Title	Authors	Year	Source Title	Citations
Assessment of primary colorectal cancer heterogeneity by using whole-tumor texture analysis [36]	Ng, F., Ganeshan, B., Kozarski, R., Miles, K.A., Goh, V.	2013	Radiology	380
Haralick texture analysis of prostate MRI: utility for differentiating non-cancerous prostate from prostate cancer [37]	Wibmer, A., Hricak, H., Gondo, T., Sala, E., Vargas, H.A.	2015	European Radiology	346
Assessment of response to tyrosine kinase inhibitors in metastatic renal cell cancer [38]	Goh, V., Ganeshan, B., Nathan, P., Vinayan, A., Miles, K.A.	2011	Radiology	327
Assessment of tumor heterogeneity by CT texture analysis: Can the largest crosssectional area be used as an alternative to whole tumor analysis? [39]	Ng, F., Kozarski, R., Ganeshan, B., Goh, V.	2013	European Journal of Radiology	316
Reproducibility of tumor uptake heterogeneity characterization through textural feature analysis in 18F-FDG PET [40]	Tixier, F., Hatt, M., Le Rest, C.C., Corcos, L., Visvikis, D.	2012	Journal of Nuclear Medicine	289
Tumour heterogeneity in oesophageal cancer assessed by CT texture analysis [41]	Ganeshan, B., Skogen, K., Pressney, I., Coutroubis, D., Miles, K.	2012	Clinical Radiology	278
Texture analysis of non-small cell lung cancer on unenhanced CT [42]	Ganeshan, B., Abaleke, S., Young, R.C.D., Chatwin, C.R., Miles, K.A.	2010	Cancer Imaging	277
Practical guidelines for choosing GLCM textures for landscape classification [43]	Hall-Beyer, M.	2017	International Journal of Remote Sensing	276
Can quantitative CT texture analysis differentiate fat-poor renal angiomyolipoma from renal cell carcinoma? [44]	Hodgdon, T., McInnes, M. D.F., Schieda, N., Lamb, L., Thornhill, R.E.	2015	Radiology	246
CT textural analysis of hepatic metastatic colorectal cancer [45]	Lubner, M.G., Stabo, N., Lubner, S.J., Halberg, R.B., Pickhardt, P.J.	2015	Abdominal Imaging	237

E. Keyword Analysis

Fig. 3 shows a network map of keyword co-occurrence of a published article on texture analysis. The keyword co-occurrence network map provides an overview of the thematic structure in entropy-based texture analysis research. Larger nodes represent frequently occurring terms while connecting lines indicate co-occurrences in published studies. Different clusters, distinguished by color, reveal key research directions. The green cluster relates to imaging techniques, encompassing terms such as “magnetic resonance imaging (MRI)”, “neural networks”, and “microstructure”. The inclusion of “high entropy alloy” suggests that entropy-based texture analysis extends beyond medical imaging, playing a role in materials science for characterising microstructural patterns. At the center of the network, “texture analysis” emerges as the most dominant keyword, closely linked with “image processing”, “feature extraction”, and “classification”, reflecting the methodological focus on extracting meaningful features for applications in medical imaging and pattern recognition. Fig. 3 illustrates the keyword co-occurrence map for entropy-based texture analysis documents, whereas Table V lists the top 10 keywords along with their total link strengths.

TABLE V. TOP 10 KEYWORDS IN ENTROPY-BASED TEXTURE ANALYSIS

Keyword	Occurrence	Link Strength
Texture Analysis	501	673
Magnetic Resonance Imaging	120	208
Radiomics	70	135
Computed Tomography	67	114
Texture	62	96
Entropy	58	87
Image Processing	38	71
Classification	35	66
GLCM	34	43
Texture Classification	34	28

The red cluster represents the core computational techniques, featuring keywords such as “entropy”, “GLCM” (grey-level co-occurrence matrix), “feature extraction”, and “wavelet transform”. The inclusion of “fractal dimension” and “Gabor filters” highlights the mathematical approaches used to quantify texture complexity. The blue cluster focuses on medical imaging applications, with terms such as “Computed

Tomography (CT)", "MRI", "PET/CT", and disease-related keywords like "lung cancer", "glioblastoma", and "metastasis", demonstrating the increasing role of entropy-based texture analysis in radiomics and diagnostic imaging. Meanwhile, the yellow and orange clusters highlight its clinical relevance, featuring keywords such as "diagnosis", "differential diagnosis", "colorectal cancer", and "gastric cancer", which emphasize its role in quantitative imaging-based diagnostics to support clinical decision-making.

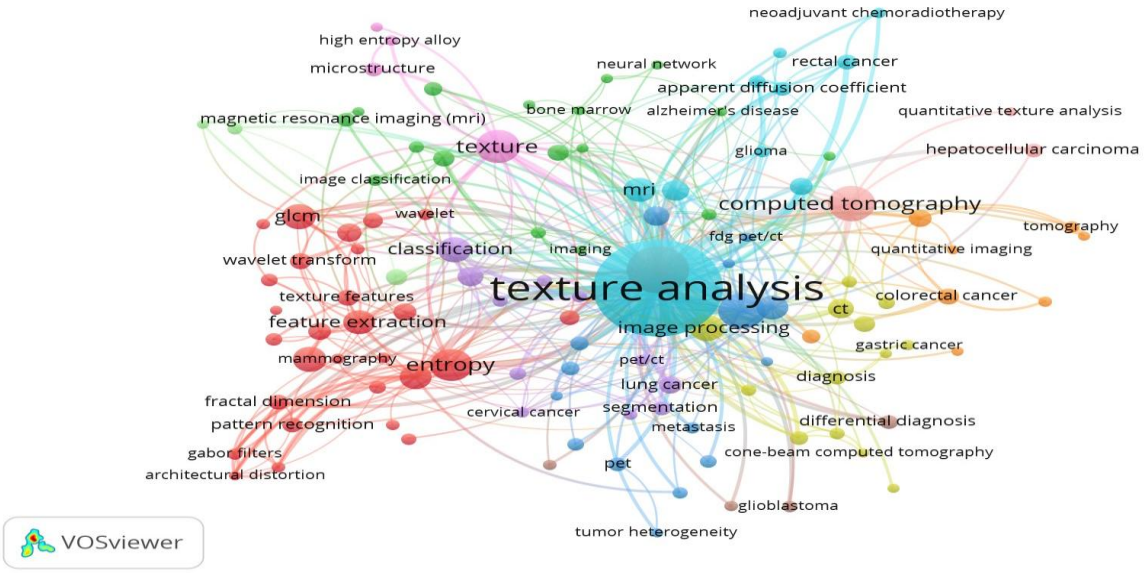


Fig. 3. Keyword co-occurrence map.

The overlay visualisation map in Fig. 4 illustrates the evolution of research trends in entropy-based texture analysis over time. Keywords shown in yellow represent emerging areas of interest, indicating the current focus of research. Recent studies highlight topics such as quantitative texture analysis, radiomics, lung cancer, MRI, computed tomography, and machine learning, suggesting a growing emphasis on advanced medical imaging techniques. In contrast, terms like entropy, feature extraction, and classification appear in green and blue, signifying their established role in the field. This progression suggests a shift from traditional texture analysis methods towards more sophisticated, imaging-driven diagnostics. Given these trends, future research is likely to further explore the integration of texture features with cutting-edge computational techniques, reinforcing their application in medical and technological advancements. The strong connections between feature extraction techniques and medical imaging terms suggest that the field is evolving towards automated and quantitative radiomics approaches. The presence of diverse methodological and application-focused clusters suggests that future research will continue refining computational techniques while expanding their use in clinical and material sciences.

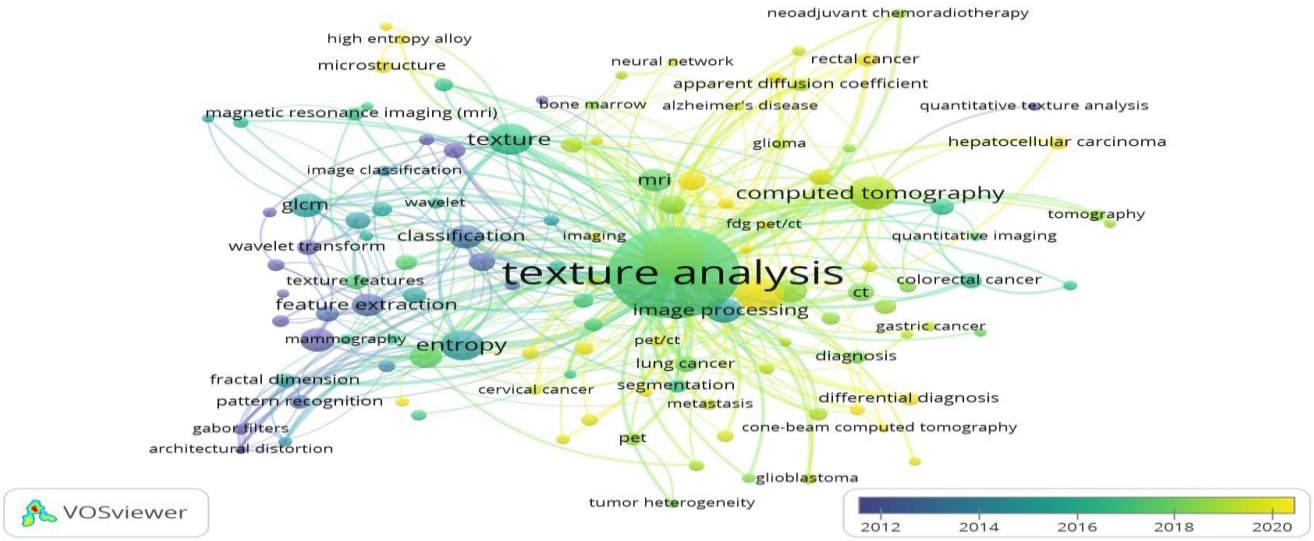


Fig. 4. Keywords overlay visualization map.

F. Geographic Contribution Analysis

The country co-authorship map presented in Fig. 5 illustrates the international collaborative landscape in the field, where each node represents a country and the lines (edges) indicate co-authorship links. The size of the node corresponds to the volume of publications, while the thickness of the links reflects the strength of collaborative ties. Countries are grouped into clusters based on the strength and frequency of their co-authorship patterns, with each cluster represented by a distinct color. Notably, China, the United Kingdom, India, and Germany appear as central hubs with extensive co-authorship ties spanning across multiple clusters, underscoring their dominant roles in facilitating global research collaboration. The green cluster, which includes countries such as India, Poland, and Saudi Arabia, represents a growing network of emerging contributors, particularly from Asia and Eastern Europe. Similarly, the red cluster, dominated by France, Brazil, Canada, and Turkey, reflects strong intercontinental research partnerships, particularly between Europe and the Americas. The blue cluster, comprising Japan, Singapore, Taiwan, and Indonesia, highlights robust regional cooperation in East and Southeast Asia. Meanwhile, the yellow cluster, centered around China and the United Kingdom, demonstrates cross-regional collaboration that bridges Asia and Europe. Countries like Italy and Mexico form part of the orange cluster, indicating active participation in Europe-Asia scientific exchanges. The inclusion of Australia and Finland in separate clusters (purple) suggests their specialised contributions and unique collaborative pathways, often acting as bridges between regions. The map's overall structure and the interconnectedness of nodes suggest a highly collaborative global research ecosystem, with developing nations increasingly integrated into international scholarly networks.

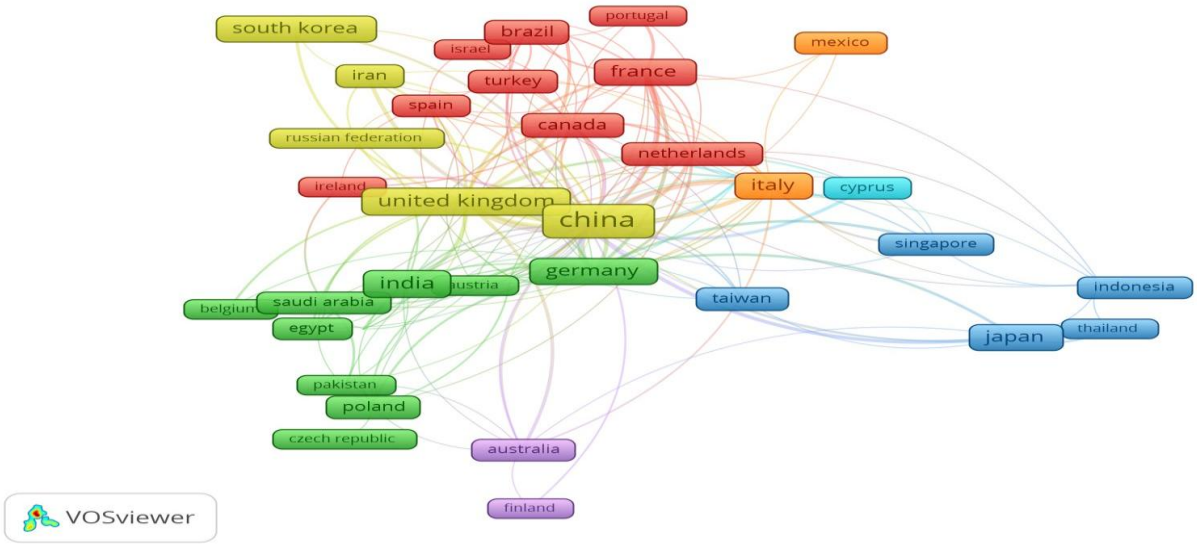


Fig. 5. Country co-authorship network.

IV. Conclusion

This bibliometric study provides a quantitative overview of entropy-based texture analysis, highlighting its growth, key contributors, and major applications. The results indicate that research in entropy-based texture analysis has experienced significant growth from 2004 to 2021, with peak activity between 2018 and 2021, followed by a decline in recent years. The findings reveal that medical imaging, particularly radiomics and tumor heterogeneity assessment, remains the dominant application area. Keyword analysis and citation mapping indicate that Shannon entropy continues to be widely used, but newer entropy measures such as sample entropy, permutation entropy, and dispersion entropy are emerging as promising techniques.

Furthermore, the study identifies strong international collaborations, with China, the United Kingdom, India, and Germany being key contributors. The interdisciplinary nature of entropy-based texture analysis is evident, spanning domains such as artificial intelligence, biomedical imaging, and material science. Despite its established applications, the field faces challenges related to standardization, computational efficiency, and interpretability of entropy measures.

A. Critical Insights and Future Directions

While this bibliometric review identifies major trends and contributors in entropy-based texture analysis, several methodological gaps remain. First, many highly cited studies still rely heavily on Shannon entropy or GLCM-based descriptors, whereas more advanced measures such as sample entropy, dispersion entropy, permutation entropy, and bidimensional variants are underutilized. This indicates a need for more comparative studies that evaluate the relative strengths and limitations of different entropy measures in diverse applications.

Second, there is a lack of standardized evaluation frameworks for entropy-based texture classification. Current studies often use different data sets, preprocessing pipelines, and classification methods, which makes direct comparisons difficult. Establishing benchmark datasets and protocols could

help ensure reproducibility and enable fair performance assessment across entropy measures.

Third, emerging application areas remain relatively unexplored. Although medical imaging dominates, entropy-based texture analysis also shows promise in materials science, industrial inspection, structural health monitoring, and security applications. For example, in industrial inspection, entropy measures can characterize surface roughness and detect micro-defects in manufacturing processes, enabling non-destructive quality control. In remote sensing, entropy-based features have been employed to classify land cover types, monitor vegetation patterns, and detect environmental changes from satellite imagery. Similarly, in biometrics, entropy-driven texture analysis has shown promise in enhancing the accuracy of fingerprint, iris, and face recognition systems by capturing fine-grained structural variations. Integrating entropy measures with deep learning and hybrid feature fusion approaches could further expand their impact, especially in radiomics and automated diagnostics.

By addressing these gaps, future research can enhance methodological robustness and broaden the scope of entropy-based texture analysis, ensuring its continued relevance in rapidly evolving technological domains. Future research should focus on improving entropy-based methods by integrating deep learning techniques, optimizing computational frameworks, and developing standardized benchmarks for texture classification. The increasing use of entropy in radiomics suggests potential for further exploration in automated diagnostics. Additionally, expanding entropy applications beyond medical imaging into remote sensing, industrial quality control, and biometrics can offer new avenues for research. Strengthening interdisciplinary collaborations will be crucial in advancing entropy-driven texture analysis and ensuring its relevance in emerging technological landscapes.

ACKNOWLEDGMENT

The author gratefully acknowledges the Postgraduate Management Center of Universiti Malaysia Terengganu for organizing a resourceful workshop on bibliometric analysis, which provided valuable guidance and tools that enabled the successful completion of this work.

DECLARATION OF GENERATIVE AI IN SCIENTIFIC WRITING

Generative AI tools were used solely to improve the readability and language of this manuscript. All AI-assisted editing was conducted under human control. The authors take full responsibility for the content of this work. No AI tools were involved in the conceptualization, analysis, interpretation of results, or drafting of scientific content. In accordance with ethical guidelines, no AI tools are listed as authors or co-authors.

CREDIT AUTHORS' STATEMENT

Muqaddas Abid: Conceptualization; Methodology; Software; Writing - Original draft preparation. Muhammad Suzuri bin Hitam: Supervision; Validation; Writing - Reviewing and Editing; Funding acquisition. Rozniza Ali: Supervision; Validation; Funding acquisition. Muhammad Hammad: Methodology; Software; Formal Analysis; Writing - Original draft preparation.

REFERENCES

[1] A. Humeau-Heurtier, "Color texture analysis: A survey," IEEE Access, vol. 10, pp. 107993–108003, 2022.

[2] Y. Liu, W.-C. Lin, and J. Hays, "Near-regular texture analysis and manipulation," ACM Trans. Graph., vol. 23, no. 3, pp. 368–376, 2004.

[3] B. J. Balas, "Texture synthesis and perception: Using computational models to study texture representations in the human visual system," Vision Res., vol. 46, no. 3, pp. 299–309, 2006.

[4] L. Armi and S. F. Ershad, "Texture image analysis and texture classification methods—A review," CoRR, abs/1904.06554, 2019.

[5] M. K. Ghalati, A. Nunes, H. Ferreira, P. Serranho, and R. Bernardes, "Texture analysis and its applications in biomedical imaging: A survey," IEEE Rev. Biomed. Eng., vol. 15, pp. 222–246, 2021.

[6] C. E. Shannon and W. Weaver, A Mathematical Model of Communication, Urbana, IL: Univ. Illinois Press, 1949, pp. 11–20.

[7] S. M. Pincus, "Approximate entropy as a measure of system complexity," Proc. Natl. Acad. Sci. USA, vol. 88, no. 6, pp. 2297–2301, 1991.

[8] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," Am. J. Physiol. Heart Circ. Physiol., vol. 278, no. 6, pp. H2039–H2049, 2000.

[9] C. Bandt and B. Pompe, "Permutation entropy: A natural complexity measure for time series," Phys. Rev. Lett., vol. 88, p. 174102, Apr. 2002.

[10] M. Rostaghi and H. Azami, "Dispersion entropy: A measure for time-series analysis," IEEE Signal Process. Lett., vol. 23, no. 5, pp. 610–614, 2016.

[11] P. Li, C. Liu, K. Li, D. Zheng, C. Liu, and Y. Hou, "Assessing the complexity of short-term heartbeat interval series by distribution entropy," Med. Biol. Eng. Comput., vol. 53, pp. 77–87, 2015.

[12] X. Liu, A. Jiang, N. Xu, and J. Xue, "Increment entropy as a measure of complexity for time series," Entropy, vol. 18, 2016.

[13] M. Costa, A. L. Goldberger, and C.-K. Peng, "Multiscale entropy analysis of complex physiologic time series," Phys. Rev. Lett., vol. 89, no. 6, p. 068102, 2002.

[14] B. García-Martínez, A. Fernández-Caballero, R. Alcaraz, and A. Martínez-Rodrigo, "Application of dispersion entropy for the detection of emotions with electroencephalographic signals," IEEE Trans. Cogn. Develop. Syst., vol. 14, no. 3, pp. 1179–1187, 2021.

[15] L. Guzmán-Vargas, A. Zabaleta-Ortega, and A. Guzmán-Sáenz, "Simplicial complex entropy for time series analysis," Sci. Rep., vol. 13, no. 1, p. 22696, 2023.

[16] G. Li, Y. Han, and H. Yang, "A new underwater acoustic signal denoising method based on modified uniform phase empirical mode decomposition, hierarchical amplitude-aware permutation entropy, and optimized improved wavelet threshold denoising," Ocean Eng., vol. 293, p. 116629, 2024.

[17] L. E. V. Silva, A. C. S. Senra Filho, V. P. S. Fazan, J. C. Felipe, and L. M. Junior, "Two-dimensional sample entropy: Assessing image texture through irregularity," Biomed. Phys. Eng. Express, vol. 2, no. 4, p. 045002, 2016.

[18] S. M. Pincus and R. R. Viscarello, "Approximate entropy: A regularity measure for fetal heart rate analysis," Obstet. Gynecol., vol. 79, no. 2, pp. 249–255, 1992.

[19] H. Azami, L. E. V. da Silva, A. C. M. Omoto, and A. Humeau-Heurtier, "Two-dimensional dispersion entropy: An information-theoretic method for irregularity analysis of images," Signal Process. Image Commun., vol. 75, pp. 178–187, 2019.

[20] H. Azami, J. Escudero, and A. Humeau-Heurtier, "Bidimensional distribution entropy to analyze the irregularity of small-sized textures," IEEE Signal Process. Lett., vol. 24, no. 9, pp. 1338–1342, 2017.

[21] M. Hilal, C. Berthin, L. Martin, H. Azami, and A. Humeau-Heurtier, "Bidimensional multiscale fuzzy entropy and its application to pseudoxanthoma elasticum," IEEE Trans. Biomed. Eng., vol. 67, no. 7, pp. 2015–2022, 2019.

[22] M. Abid, M. S. Hitam, R. Ali, H. Azami, and A. Humeau-Heurtier, "Bidimensional increment entropy for texture analysis: Theoretical validation and application to colon cancer images," Entropy, vol. 27, no. 1, p. 80, 2025.

[23] A. S. Gaudêncio, M. Hilal, J. M. Cardoso, A. Humeau-Heurtier, and P. G. Vaz, "Texture analysis using two-dimensional permutation entropy and amplitude-aware permutation entropy," Pattern Recognit. Lett., vol. 159, pp. 150–156, 2022.

[24] W. Li, Y. Zhao, Q. Wang, and J. Zhou, "Twenty years of entropy research: A bibliometric overview," Entropy, vol. 21, no. 7, p. 694, 2019.

[25] W. H. Lam, W. S. Lam, S. H. Jaaman, and P. F. Lee, "Bibliometric analysis of information theoretic studies," Entropy, vol. 24, no. 10, p. 1359, 2022.

[26] N. J. Van Eck and L. Waltman, "Citation-based clustering of publications using CitNetExplorer and VOSviewer," Scientometrics, vol. 111, pp. 1053–1070, 2017.

[27] N. Pakniyat, J. Abdullah, O. Krejcar, and H. Namazi, "Complexity-based analysis in biomedical image analysis: A review," Fractals, vol. 32, no. 6, p. 2430002, 2024.

[28] T. Akram, A. Alsuhaibani, M. A. Khan, S. U. Khan, S. R. Naqvi, and M. Bilal, "Dermo-optimizer: Skin lesion classification using information-theoretic deep feature fusion and entropy-controlled binary bat optimization," Int. J. Imaging Syst. Technol., vol. 34, no. 5, p. e23172, 2024.

[29] M. Abid, M. S. Hitam, R. Ali, H. Azami, and A. Humeau-Heurtier, "Spatio-frequency texture analysis using wavelet increment entropy: Methodology and application to MRI in multiple sclerosis," Inf. Sci., p. 122669, 2025.

[30] J. Mahmoodi, D. Abbasi-Moghadam, A. Sharifi, H. Nezamabadipour, M. Esmaeili, and A. Vafaeinejad, "Dessa-net model: Hyperspectral image classification using an entropy filter with spatial and spectral attention modules on deepnet," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., 2024.

[31] Y. Hua, X. Xi, C. Qu, J. Du, M. Weng, and B. Ye, "An adaptive watermarking for remote sensing images based on maximum entropy and discrete wavelet transformation," KSII Trans. Internet Inf. Syst., vol. 18, no. 1, pp. 192–210, 2024.

[32] S. Agarwal and K.-H. Jung, "Digital image steganalysis using entropy-driven deep neural network," J. Inf. Secur. Appl., vol. 84, p. 103799, 2024.

[33] A. A. Saddek, T.-K. Lin, Y.-T. Lin, and K.-W. Kuo, "Enhancing the credibility of an entropy-based structural health monitoring system using convolutional neural networks," J. Civil Struct. Health Monit., pp. 1–17, 2025.

[34] F. Abdollahi-Mamoudan, C. Ibarra-Castanedo, and X. P. V. Maldague, "Non-destructive testing and evaluation of hybrid and advanced structures: A comprehensive review of methods, applications, and emerging trends," Sensors, vol. 25, no. 12, p. 3635, 2025.

[35] K. An, C. Li, G. Long, and J. Ding, "An improved entropy regularization method for defect detection in multi-layer composites," IEEE Trans. Antennas Propag., 2024.

[36] F. Ng, B. Ganeshan, R. Kozarski, K. A. Miles, and V. Goh, "Assessment of primary colorectal cancer heterogeneity by using whole-tumor texture analysis: Contrast-enhanced CT texture as a biomarker of 5-year survival," Radiology, vol. 266, no. 1, pp. 177–184, 2013.

[37] A. Wibmer, H. Hricak, T. Gondo, K. Matsumoto, H. Veeraraghavan, D. Fehr, J. Zheng, D. Goldman, C. Moskowitz, S. W. Fine, et al., "Haralick texture analysis of prostate MRI: Utility for differentiating noncancerous prostate from prostate cancer and differentiating prostate cancers with different Gleason scores," Eur. Radiol., vol. 25, pp. 2840–2850, 2015.

[38] V. Goh, B. Ganeshan, P. Nathan, J. K. Juttla, A. Vinayan, and K. A. Miles, "Assessment of response to tyrosine kinase inhibitors in metastatic renal cell cancer: CT texture as a predictive biomarker," Radiology, vol. 261, no. 1, pp. 165–171, 2011.

[39] F. Ng, R. Kozarski, B. Ganeshan, and V. Goh, "Assessment of tumor heterogeneity by CT texture analysis: Can the largest cross-sectional area be used as an alternative to whole tumor analysis?" Eur. J. Radiol., vol. 82, no. 2, pp. 342–348, 2013.

[40] F. Tixier, M. Hatt, C. C. Le Rest, A. Le Pogam, L. Corcos, and D. Visvikis, "Reproducibility of tumor uptake heterogeneity characterization through textural feature analysis in 18F-FDG PET," J. Nucl. Med., vol. 53, no. 5, pp. 693–700, 2012.

[41] B. Ganeshan, K. Skogen, I. Pressney, D. Coutroubis, and K. Miles, "Tumour heterogeneity in oesophageal cancer assessed by CT texture analysis: Preliminary evidence of an association with tumour metabolism, stage, and survival," Clin. Radiol., vol. 67, no. 2, pp. 157–164, 2012.

[42] B. Ganeshan, S. Abaleke, R. C. D. Young, C. R. Chatwin, and K. A. Miles, "Texture analysis of non-small cell lung cancer on unenhanced computed tomography: Initial evidence for a relationship with tumour glucose metabolism and stage," Cancer Imaging, vol. 10, no. 1, p. 137, 2010.

[43] M. Hall-Beyer, "Practical guidelines for choosing GLCM textures to use in landscape classification tasks over a range of moderate spatial scales," Int. J. Remote Sens., vol. 38, no. 5, pp. 1312–1338, 2017.

[44] T. Hodgdon, M. D. F. McInnes, N. Schieda, T. A. Flood, L. Lamb, and R. E. Thornhill, "Can quantitative CT texture analysis be used to differentiate fat-poor renal angiomyolipoma from renal cell carcinoma on unenhanced CT images?" Radiology, vol. 276, no. 3, pp. 787–796, 2015.

[45] M. G. Lubner, N. Stabo, S. J. Lubner, A. Munoz Del Rio, C. Song, R. B. Halberg, and P. J. Pickhardt, "CT textural analysis of hepatic metastatic colorectal cancer: Pre-treatment tumor heterogeneity correlates with pathology and clinical outcomes," Abdom. Imaging, vol. 40, pp. 2331–2347, 2015.

[46] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," IEEE Trans. Syst. Man Cybern., no. 6, pp. 610–621, 1973.

[47] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," Pattern Recognit., vol. 29, no. 1, pp. 51–59, 1996.

[48] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," Pattern Recognit., vol. 24, no. 12, pp. 1167–1186, 1991.

A Review of Visualization Techniques for Duplicate Detection in Cancer Datasets

Nurul A. Emran[1]*, Ruhaila Maskat[2]
Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, 76100, Melaka, Malaysia [1]
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450, Selangor, Malaysia[2]

***Abstract*—As clinical cancer research increasingly depends on large, diverse datasets, concerns about data duplication have grown. Duplicates can undermine data integrity, skew analytical results, and reduce the reproducibility of studies. This review explores how visualization can play a critical role in identifying and managing duplicates in non-image clinical cancer data. Drawing from literature in biomedical informatics, data quality, and visual analytics, it synthesizes current approaches and highlights key challenges. Using a scoping review methodology, we analyzed studies published over the past two decades, focusing on non-image clinical datasets. Studies were selected based on relevance to duplicate detection and visualization, excluding those centered on image or video data. Major datasets like The Cancer Genome Atlas (TCGA), The Cancer Imaging Archive (TCIA), and the North American Association of Central Cancer Registries (NAACCR) are examined to show how duplication occurs across genomic, clinical, and registry data. The review assesses existing visualization techniques based on their scalability, interactivity, integration with deduplication algorithms, and how well they address core data quality dimensions. While some tools offer scalable and interactive features, few provide clear visual representations of duplicates, especially those involving complex temporal and multidimensional patterns. Several methodological gaps are identified, including limited integration of data quality metrics, inadequate support for tracking changes over time, and a lack of standardized evaluation frameworks. To address these issues, the review advocates for the development of practical, user-friendly visualization tools that combine duplicate detection with key indicators of data quality. By offering a more complete and intuitive view of clinical datasets, such tools can help researchers and clinicians make better-informed decisions, ultimately improving the reliability and impact of cancer research. Bridging the gap between technical detection and visual understanding is essential for advancing data-driven healthcare and ensuring high-quality, reproducible outcomes.**

***Keywords*—Duplicate detection; data duplication; visualization; deduplication; TCGA; TCIA; NAACCR**

I. Introduction

With the growing volume and complexity of biomedical data, especially in clinical cancer research, the issue of duplicate records has become increasingly important. Ensuring high data quality is essential, as duplication can significantly affect research accuracy and outcomes. As a result, visualizing duplicates in these datasets has emerged as a crucial area of study [1], [2]. In the past decade, improvements in data collection and integration have produced large, diverse clinical datasets. These datasets demand advanced visualization techniques to effectively assess data quality and support informed clinical decision-making [3], [4], [5], [6]. Duplicate records, where the same real-world entity appears multiple times can create serious challenges in data analysis. They may artificially boost predictive performance and lead to misleading research conclusions, ultimately compromising the reliability of clinical insights [1], [7]. Numerous studies have shown that duplicate records are common across various biomedical databases, including those containing transcriptomic and nucleotide sequence data. This highlights an urgent need for more effective methods to detect and visualize duplicates, ensuring data integrity and improving research outcomes [7]. Studies have shown that duplicate records can account for up to 30–50% of entries in some clinical databases, leading to inflated sample sizes, biased statistical outcomes, and compromised patient safety [8], [9]. The practical impact of duplicate records is clear as studies show they can lead to redundant or conflicting results, which in turn can disrupt downstream analyses and misguide clinical interpretations [7], [10]. This review focuses on a specific yet underexplored challenge: visualizing duplicate records in non-image clinical cancer datasets. Despite its importance, current methods in this area remain fragmented and underdeveloped, limiting their effectiveness in real-world clinical applications [1], [11].

Although many algorithms exist for detecting duplicates and assessing data quality, there's still a noticeable gap when it comes to visualization approaches specifically designed for the unique complexities of clinical cancer data. Current solutions often fall short in providing comprehensive, intuitive visual tools that support meaningful analysis in this domain [12], [13], [14]. There are differing views on the best approaches for identifying and representing duplicates, ranging from statistical record linkage methods to machine learning-based similarity measures. However, few studies have successfully combined these techniques with visualization tools that support human-in-the-loop data quality assessment, limiting their practical usefulness in clinical settings [15]–[17]. This gap can lead to serious consequences, including misinterpreting data, wasting time on inefficient curation efforts, and drawing flawed clinical insights that may affect patient care and research outcomes [7], [18]. Given the wide variety of data types and the complexity of clinical workflows, there's a clear need for visualization tools that can effectively handle multidimensional, temporal, and categorical data. These tools must be capable of adapting to the intricate nature of clinical cancer datasets to support accurate analysis and decision-making [19], [20].

This review introduces a conceptual framework that defines duplicates as multiple records referring to the same clinical

*Corresponding Author.

entity. It connects this definition to key data quality dimensions, such as completeness, accuracy, and consistency to emphasize the broader impact of duplication on clinical data integrity [12], [21]. Visualization is framed as a guided, interactive process that blends computational techniques for detecting duplicates with human pattern recognition. This approach enables users to actively assess and manage data quality, making the process more intuitive and effective [12], [22].

The framework brings together theories of data quality assessment and principles of visualization system design, highlighting how visual analytics can play a key role in managing clinical oncology data more effectively [3], [23]. This review aims to examine how current visualization tools and techniques are being applied to detect and represent duplicate records in clinical cancer datasets particularly those that do not involve image or video data. By drawing insights from fields such as biomedical informatics, data quality, and visual analytics, the review seeks to identify gaps in existing approaches and highlight opportunities for advancement.

A key contribution of this work is its effort to bridge the divide between technical duplicate detection algorithms and practical visualization tools that help researchers and clinicians interpret and manage data more effectively. This integration supports improved data curation and enhances the reliability of cancer research outcomes. To guide this exploration, the review adopts a scoping methodology, analyzing studies published over the past two decades. Literature searches were conducted across major academic databases, including PubMed, IEEE Xplore, Scopus, and ACM Digital Library, to ensure comprehensive coverage of relevant work.

The review focuses specifically on research related to duplicate detection and visualization in clinical or biomedical datasets, excluding studies centered on image or video data. This review deliberately excludes image and video data to focus on structured and semi-structured clinical datasets, such as EHRs, genomic records, and registries, where duplication manifests differently and is often harder to visualize. While this boundary narrows the scope, it allows for a more focused analysis of underexplored challenges in non-visual data domains, which are critical for data-driven decision-making in clinical workflows. The findings are organized to first outline the types of visualization techniques currently in use, then discuss the challenges and available tools, and finally propose directions for future research.

The remainder of the paper is organized as follows: Section II provides an overview of visualization approaches used in duplicate detection within clinical cancer datasets, focusing on technique diversity, data quality coverage, scalability, and user interaction. Section III presents a critical evaluation of methodological strengths and limitations, emphasizing integration with deduplication algorithms and temporal visualization. Section IV discusses the theoretical and practical implications of duplicate visualization, emphasizing the need for adaptive, context-aware frameworks and scalable visual tools. It highlights the integration of temporal stability and multidimensional analytics, advocating for human-in-the-loop systems that enhance data quality, support clinical decision-making, and improve research reliability. Section V introduces key datasets relevant to duplicate detection and visualization, discussing their characteristics and challenges. Section VI outlines existing gaps and proposes future research directions to advance the field. Finally, Section VII concludes the paper by summarizing key findings and highlighting the importance of integrating visualization with data quality assessment to improve clinical research outcomes.

II. Overview of Visualization Approaches in Duplicate Detection

This section delves into how researchers are currently addressing the challenge of visualizing duplicate records in clinical cancer datasets, particularly those that exclude image and video data. It explores a broad spectrum of tools, techniques, and approaches, some focused on developing new solutions, others aimed at evaluating existing systems. The reviewed studies range from interactive visual analytics platforms to scalable computational frameworks, all designed to manage the complexity of large, heterogeneous clinical datasets. To assess the effectiveness of these approaches, the review considers several key factors: the diversity of visualization techniques employed, their ability to address various dimensions of data quality, scalability to large datasets, level of user interactivity, and integration with duplicate detection algorithms. By comparing these elements, the review identifies both strengths and areas where further development is needed. A central takeaway is the pressing need for more comprehensive, user-centered visualization tools, ones that not only detect duplicates but also help users understand their broader impact on data quality. This overview sets the stage for future innovation, emphasizing the importance of creating tools that are not only technically robust but also practical and relevant for clinical use. A summary table in Table I is included to show how different studies map across these key dimensions.

A. Visualization Technique Diversity

A variety of studies have applied diverse visualization techniques, including flow diagrams, glyph-based representations, set visualizations, timelines, and interactive dashboards, to support duplicate detection and assess data quality in clinical cancer and broader biomedical datasets [1], [23], [24]. While basic formats such as tables and bar charts remain prevalent, recent surveys indicate growing interest in omics-driven and multidimensional visualizations tailored to the complexity of clinical data [3], [25]. Novel visualization techniques such as Missingness Glyphs and ensemble glyphs have been proposed to enhance pattern recognition in missing data and ensemble comparisons, respectively [26], [27]. Comparative visualization approaches for temporal and multivariate data have been developed to support patient cohort analysis and similarity assessments [20], [28].

B. Data Quality Dimension Coverage

Many studies have addressed multiple dimensions of data quality such as completeness, accuracy, consistency, and temporal stability, with particular emphasis on temporal aspects, which are especially critical in clinical and biomedical contexts [24], [29], [30]. Completeness and accuracy were the most commonly visualized data quality dimensions, particularly in studies focused on analyzing missing data patterns and detecting duplicate records [17], [21]. Temporal stability and changes over

time were captured using probabilistic change detection methods and time-oriented visualizations, allowing researchers to monitor how data quality evolves across clinical datasets [20], [30]. Some studies concentrated on specific data quality dimensions, for example, focusing on consistency in record linkage or accuracy in similarity calculations used for deduplication tasks [14], [15], [31].

C. Scalability and Performance

Scalability emerged as a key concern, with several studies showcasing visualization and deduplication techniques capable of processing millions of records, including complex datasets like clinical notes and adverse drug reaction databases [32], [33]. To improve scalability in duplicate detection pipelines, several studies leveraged parallel and distributed computing frameworks such as Apache Spark and cluster-based algorithms enabling efficient processing of large-scale clinical datasets [33], [34]. The scalability of visualization tools varied widely across studies. While some were built to handle large, heterogeneous clinical datasets, others were limited to smaller or synthetic datasets, restricting their practical application in real-world clinical environments [12], [19]. Balancing detail and overview in large datasets often involved trade-offs. To manage this, several studies used techniques like hierarchical clustering and level-of-detail visualizations, helping users navigate complex data without losing sight of important patterns or broader trends [22],[30].

D. User Interaction and Interpretability

Many visualization systems supported interactive exploration, allowing domain experts to actively engage with the data spotting anomalies, uncovering patterns in missing data, and comparing patient cohorts with greater ease and insight [15], [18], [19]. Some tools offered only limited interactivity, placing greater emphasis on automated detection and algorithmic outputs. As a result, user engagement was minimal, reducing opportunities for experts to explore data patterns or validate findings through direct interaction [1], [27].

Visual analytics approaches often incorporated user feedback and domain expertise to improve the clarity and trustworthiness of data quality assessments. By involving experts in the process, these tools became more interpretable and aligned with real-world clinical needs [4], [18]. Set-based and glyph visualizations stood out for their ability to make complex data quality issues easier to understand. Their intuitive design helped users quickly grasp patterns and anomalies, making them valuable tools for exploring duplicate records and other data inconsistencies [19], [21].

E. Integration with Deduplication Algorithms

There is a consensus that integrating visualization with deduplication and data cleansing algorithms enhances detection accuracy and facilitates iterative refinement of analytical processes [8], [12], [27], [30]. Certain frameworks integrate similarity computations directly with visualization components to provide deeper analytical insights and improve interpretability [10], [31]. Several tools offer limited interactivity, prioritizing automated detection and algorithmic outputs over active user involvement [1], [27].

Building upon this descriptive overview, the next section provides a critical analysis of the reviewed studies, highlighting their strengths, limitations, and methodological gaps. This synthesis aims to offer a clearer understanding of current practices and to outline future directions for duplicate visualization in clinical cancer datasets.

III. Evaluation of Methodological Strengths and Limitations

The existing literature on duplicate visualization in clinical cancer datasets presents a wide array of methodologies and tools, reflecting both innovative advancements and persistent challenges. Notable strengths include the development of scalable algorithms and interactive visualization techniques capable of handling complex, heterogeneous data. However, several limitations remain, particularly in the integration of data quality metrics, temporal dynamics, and user-centered design. These issues are often exacerbated by the experimental nature of many tools and the lack of comprehensive evaluations. This synthesis highlights the pressing need for more robust, sustainable, and clinically integrated visualization solutions that can effectively support duplicate detection and data quality assessment in non-image-based clinical cancer data.

A. Visualization Techniques for Duplicate Detection

A number of studies introduce innovative visualization techniques specifically designed for complex clinical datasets. These include interactive set visualizations to explore patterns of missing data and glyph-based methods to highlight data quality issues. Such approaches help users more effectively identify duplicates and anomalies within large, heterogeneous datasets [8], [19], [21]. Techniques such as Minhashing combined with Locality Sensitive Hashing have shown strong scalability, making it possible to efficiently detect duplicates within large volumes of clinical notes [27]. Despite notable progress, many visualization techniques remain insufficiently developed for the explicit detection of duplicates in clinical cancer datasets, especially those involving non-image data. The continued reliance on basic formats such as tables often limits both the interpretability and effectiveness of identifying duplicate records [3]. Moreover, the absence of standardized evaluation metrics and the limited number of comprehensive user studies hinder the ability to rigorously assess the effectiveness of these visualization methods [3], [18].

B. Integration of Data Quality Metrics

Some studies incorporate key data quality dimensions, such as completeness, accuracy, and consistency into their visualization frameworks, allowing users to more effectively identify and interpret data defects and duplicate entries [2], [8], [16]. Probabilistic frameworks for assessing temporal stability offer novel approaches to visualizing changes in data quality over time, which is particularly important for longitudinal clinical datasets [25]. However, the integration of data quality metrics into duplicate visualization tools is often limited or implicit. Many existing tools fall short of explicitly combining multiple dimensions of data quality or providing comprehensive visual analytics that address both temporal and multidimensional aspects of duplicate detection [3], [16]. The inherent complexity of clinical data, coupled with the diverse

nature of data quality issues, presents challenges that current visualization systems have yet to fully address [18].

C. Temporal and Multidimensional Visualization

Recent research has explored temporal visualization techniques such as timelines and temporal line charts to capture evolving patterns in patient data, which are relevant for understanding how duplicates emerge over time [15], [25]. Multidimensional approaches, including flow visualizations and hierarchical clustering, have also been applied to support the exploration of complex cancer registry and ensemble datasets [14], [22]. However, the explicit visualization of temporal dynamics and multidimensional characteristics of duplicates remains limited. Few systems enable direct comparisons between individual patients and cohorts or incorporate temporal stability assessments into duplicate detection workflows [15]. The high dimensionality and sparsity of clinical data further complicate effective temporal visualization, often resulting in oversimplified or fragmented representations [25].

D. Scalability and Performance of Visualization Tools

Several studies have introduced scalable algorithms and parallel processing techniques, such as Spark-based kNN classifiers and parallel deduplication methods that enable efficient handling of large clinical datasets containing millions of record [28], [29]. These approaches support real-time or near-real-time duplicate detection and visualization, which are essential for clinical applications. However, despite these advancements in scalability, many visualization tools face challenges in maintainability and lack ongoing updates, often leading to their discontinuation after initial publication [3]. The transition from prototype to routine clinical use is further hindered by resource limitations and insufficient software engineering practices, ultimately reducing the practical impact of these tools [3].

TABLE I. KEY DIMENSIONS OF STUDIES IN DUPLICATE VISUALIZATIONS

KEY DIMENSIONS	DESCRIPTIONS	STUDIES
Visualization Technique Diversity	This dimension explores how researchers visually represent duplicate records and related data quality issues in cancer datasets. Some rely on straightforward visuals like tables and bar charts, while others use more advanced methods such as glyphs, set-based diagrams, flow charts, and time-based plots. A diverse range of visualization styles allows researchers to examine duplicates from multiple perspectives, whether it's structural patterns, temporal changes, or categorical overlaps, making it easier to identify problems and gain clearer insights into the data.	[1], [15], [18], [19], [20], [26], [27]
Data Quality Dimension Coverage	This dimension focuses on how effectively visualization tools help users understand different aspects of data quality, such as completeness, accuracy, consistency, and how stable the data is over time. Rather than simply pointing out duplicates, well-designed visualizations show how those duplicates impact the overall reliability of the dataset. When multiple quality indicators are presented together, researchers gain a clearer and more comprehensive view of the data's trustworthiness.	[10],[12], [14], [15], [16], [19], [24], [25], [26]
Scalability and Performance	This dimension focuses on how well a visualization tool can manage large and complex cancer datasets, which often contain millions of records. A good system should remain responsive and easy to use, even when handling massive amounts of data. In real-world clinical environments, where data is often messy and highly varied, having a tool that performs smoothly and scales effectively is essential for making timely and accurate decisions.	[8],[14],[22],[27],[28],[29], ,[30]
User Interaction and Interpretability	This dimension focuses on how easily users can interact with visualization tools to explore and understand duplicate records. Features like zooming, filtering, comparing patient groups, and highlighting patterns allow users to engage with the data in meaningful ways. When visualizations are intuitive and responsive, they help researchers and clinicians quickly spot issues, ask relevant questions, and draw informed conclusions based on what they see.	[1], [4],[5], [6],[20], [21], [23], [24], [27],
Integration with Deduplication Algorithms	This dimension looks at how well visualization tools are integrated with computational methods for detecting duplicates, such as record linkage, similarity scoring, or machine learning. When these tools are tightly connected, users can easily visualize algorithm outputs, validate results, and adjust detection settings interactively. This integration helps bridge the gap between automated data processing and expert-driven data curation, making the entire process more transparent and user-friendly.	[1], [8], [10],[12], [27], [30], [31], [32]

E. User Interaction and Interpretability

Interactive visualization features such as dynamic filtering, selection mechanisms, and focus+context techniques play a key role in enhancing user engagement and enabling detailed exploration of duplicates and data quality issues [3], [23], [28]. Visual analytics systems that integrate data cleansing with exploratory workflows empower domain experts to iteratively uncover anomalies and patterns [23]. However, many existing tools offer only limited interactivity, often confined to basic search or hover functions, which may be inadequate for addressing the complexity of duplicate detection tasks [3]. The diversity and flexibility of visualization and user interface components also complicate usability evaluations and can hinder user adoption, particularly in multidisciplinary clinical environments. Additionally, the visualization of patient-reported outcomes and follow-up data remains an underexplored area.

F. Methodological Robustness and Evaluation

Some studies validate their duplicate detection and visualization methods through rigorous experimental evaluations, including benchmarking against large, well-curated duplicate datasets and incorporating expert assessments [1], [36], [37]. The use of probabilistic models and adaptive similarity measures has improved detection accuracy and allowed methods to better align with domain-specific characteristics of clinical data [15], [17]. However, the overall methodological rigor across studies varies considerably. Many lack comprehensive validation or rely on limited datasets, which undermines the reliability of their findings. The absence of standardized benchmarks for clinical cancer datasets and inconsistent reporting of visualization capabilities further complicate comparative evaluations and limit generalizability [3], [11]. Additionally, the inherent complexity of clinical data introduces confounding factors that challenge the robustness and consistency of duplicate detection algorithms [31].

G. Opportunities for Innovation

The literature highlights clear gaps and opportunities for advancing visualization strategies that integrate temporal stability, multidimensional data quality metrics, and user-centered interaction to improve duplicate detection and data quality assessment [3], [17], [34]. There is growing recognition of the potential to combine computational techniques with visual analytics to enhance interpretability and support clinical decision-making [4], [5], [6], [17]. Major challenges include the diversity of data types, the need for scalable solutions capable of handling large datasets, and the difficulty of embedding these tools into real-world clinical workflows. Addressing these issues will require stronger interdisciplinary collaboration yet such efforts are still relatively limited.

Across the reviewed studies, there is a broad consensus that visualization plays a vital role in helping researchers and clinicians detect and interpret duplicates within cancer and biomedical datasets. Most agree on the importance of incorporating multiple data quality indicators such as completeness and temporal stability and ensuring that tools are capable of managing large, complex datasets.

However, perspectives diverge on the extent to which visualization tools should be tightly integrated with deduplication algorithms, the diversity of visual techniques to be employed, and the level of user interaction required. These differences often reflect the distinct needs of various research domains, data types, and technical approaches, ranging from genetic sequence databases to clinical notes and electronic health records. Together, these shared understandings and differing viewpoints contribute to a more comprehensive and nuanced picture of the current landscape. They not only highlight the complexities and challenges involved in duplicate visualization within clinical cancer data, but also illuminate emerging trends, gaps in existing methodologies, and opportunities for innovation. This synthesis of perspectives serves as a foundation for identifying strategic directions for future research, refining theoretical frameworks, and enhancing the practical implementation of visualization techniques in clinical oncology settings.

IV. Theoretical and Practical Implications

This review highlights that duplicates in clinical cancer datasets are often complex and nuanced. Rather than simple repeated entries, duplicates may contain conflicting information or evolve over time, making them difficult to detect and interpret. Their inconsistent nature and varied impact on data analysis underscore the need for flexible, context-aware frameworks that can accommodate different types of duplicates and their implications for research and clinical decision [17], [36], [37]. Studies show that employing adaptive similarity measures particularly those capable of learning from the data can significantly enhance detection accuracy. These approaches move beyond traditional techniques like basic text matching, aligning with theoretical advancements in metric functional dependencies and trainable similarity functions [15], [17], [38].

The integration of temporal stability as a data quality dimension introduces a novel theoretical lens, emphasizing the dynamic nature of clinical data and the importance of probabilistic and information-geometric methods for identifying and characterizing temporal changes in duplicates and data quality [30]. Visualization theory is also evolving, recognizing the critical role of interactive and scalable visual analytics in managing heterogeneous clinical data. These tools support hypothesis generation and enable exploration of the multidimensional and temporal aspects of duplicates [19], [22], [28]. Furthermore, the literature challenges the adequacy of single benchmark evaluations, advocating for the use of multiple, diverse, and validated benchmarks to ensure the robustness and generalizability of duplicate detection models [36], [37]. Visual analytics frameworks are increasingly enriched by human-in-the-loop approaches, where visualization serves as a bridge between complex algorithmic outputs and user interpretation particularly in high-stakes settings such as clinical oncology and tumor board decision-making [3], [22].

For both clinical research and healthcare practice, the findings underscore the importance of making duplicate detection and visualization routine components of data management. These processes are essential for maintaining data quality, avoiding misleading analyses, and supporting reliable clinical decision-making, as demonstrated by tools like doppelgangR and integrated visual cleansing workflows [1], [23]. The successful deployment of scalable and parallelized algorithms [35], including those based on Minhashing, Locality Sensitive Hashing, and Spark-based kNN classifiers, illustrates the feasibility of handling large-scale clinical datasets, which is critical for real-world applications in hospital and research environments [32], [33]. Visualization tools that incorporate multidimensional, temporal, and set-based techniques enable practitioners to uncover complex patterns of duplication and missing data, thereby facilitating targeted data quality interventions and enhancing the interpretability of clinical datasets [24], [26], [39].

The adoption of interactive visualization systems that support user-driven exploration, such as temporal stability assessments and cohort comparisons has the potential to significantly enhance tumor board workflows and multidisciplinary clinical decision-making processes [3], [20]. To support this, industry and policy stakeholders should prioritize the development of comprehensive data quality monitoring infrastructures that incorporate visualization components, enabling continuous tracking and traceability of duplicates and other data quality issues within clinical data warehouses [2]. Evidence suggests that advanced visualization and duplicate detection techniques not only improve research accuracy but also contribute to patient safety, streamline healthcare operations, and support regulatory compliance. These benefits underscore the importance of investing in robust data quality tools across healthcare systems [16].

While existing visualization taxonomies often focus on general data exploration or image-based analytics, this review introduces a conceptual framework specifically tailored to the visualization of duplicate records in non-image clinical cancer datasets. Unlike traditional models that treat data quality and visualization as separate domains, our framework integrates duplicate detection algorithms, interactive visual analytics, and data quality dimensions (e.g. completeness, temporal stability) into a unified structure. This approach emphasizes human-in-

the-loop interpretation, temporal dynamics, and multidimensional data contexts areas that are underrepresented in current taxonomies. By aligning visualization strategies with specific data quality challenges posed by duplication, the framework offers a more targeted lens for evaluating and designing tools in clinical research settings.

The findings from this section further emphasize the complexity and variability of duplicate records in clinical cancer datasets. Addressing these challenges effectively requires well-structured datasets and adaptable tools capable of handling diverse duplication scenarios. To support this effort, the following section introduces key datasets that can aid researchers in developing more effective strategies for detecting and visualizing duplicates in cancer data.

V. Key Datasets for Duplicate Detection and Visualization

A range of publicly available datasets serve as valuable resources for researchers working to detect and visualize duplicates in clinical cancer data. These assets provide a foundational platform for developing more effective methods to clean, interpret, and analyze complex biomedical information:

1) The Cancer Genome Atlas (TCGA) [1] *:* The Cancer Genome Atlas (TCGA), accessible via the Genomic Data Commons (GDC), stands as one of the most comprehensive resources for cancer genomics. It integrates clinical and molecular data from 33 cancer types, encompassing thousands of patient cases and millions of data files. With its diverse range of data including whole genome sequencing, RNA-seq, methylation, and proteomic profiles, TCGA offers a powerful foundation for cancer research.

However, the scale and complexity of TCGA also present challenges for duplicate detection. The involvement of multiple contributors and data modalities increases the risk of overlapping entries, such as repeated patient identifiers or biospecimen records. While the GDC mitigates some of these risks through standardized clinical data formats and consistent bioinformatics workflows, duplicate detection remains a critical step to ensure data integrity and reliability.

TCGA also provides interactive visualization tools, such as the Cohort Builder and Mutation Frequency plots, which support the exploration of genomic alterations and clinical attributes. These tools are particularly useful for identifying anomalies or inconsistencies that may signal duplicate records, especially when combined with metadata filters and cohort analysis. Overall, TCGA is a cornerstone of cancer research, and its structured data along with integrated visualization capabilities make it an essential resource for addressing duplication challenges in large-scale biomedical datasets.

2) The Cancer Imaging Archive (TCIA) [2] *:* The Cancer Imaging Archive (TCIA) is a publicly accessible repository offering a vast collection of de-identified medical images related to cancer. These datasets are organized into collections based on disease types, such as lung cancer and imaging modalities, including MRI, CT scans, and histopathology. TCIA supports a variety of formats, with Digital Imaging and Communications in Medicine (DICOM) serving as the standard for radiological data.

Due to its aggregation of data from multiple institutions, TCIA presents both opportunities and challenges for duplicate detection. The diversity of contributors and imaging types increases the likelihood of overlapping records, such as repeated patient identifiers, imaging sessions, or processed files. Fortunately, TCIA provides rich metadata, including treatment details, patient outcomes, and genomic annotations, that can assist researchers in identifying and resolving potential duplicates.

The platform also features visualization tools and filtering options that enable users to explore imaging datasets and detect anomalies or inconsistencies. These capabilities are particularly useful for identifying redundant scans or mismatched patient data, which could compromise the accuracy of research findings. Overall, TCIA is a valuable resource for cancer imaging research, and when paired with visualization and record linkage techniques, it supports effective deduplication and enhances data quality.

3) North American Association of Central Cancer Registries (NAACCR) CiNA[3]*:* The NAACCR Cancer in North America (CiNA) Research dataset is a robust, population-based resource that consolidates cancer data from registries across the United States and Canada. It offers detailed insights into cancer incidence, survival, and prevalence, and is curated to meet high standards of completeness and data quality, making it an essential tool for cancer surveillance and public health research. While CiNA provides significant value, duplicate detection remains a challenge due to the integration of data from multiple state and provincial registries. Patients who move across regions or receive care from different institutions may have overlapping records. NAACCR addresses this through standardized coding practices and routine quality checks, but researchers must still exercise caution, particularly when working with multi-year or multi-registry datasets.

CiNA's rich metadata, including patient demographics, diagnosis timelines, and tumor characteristics, supports effective deduplication. Tools like SEER*Stat enable researchers to visualize and analyze these variables, helping to identify inconsistencies or repeated entries. This is especially important when linking CiNA with other datasets or conducting longitudinal studies, where duplicate records can distort findings. Overall, CiNA is a high-quality dataset that, when paired with appropriate visualization and data linkage techniques, facilitates accurate duplicate detection and enhances data reliability.

4) KAUH-BCMD[4]*:* The KAUH-BCMD dataset, developed by Al-Mnayyis et al. [40], comprises over 7,000

[1] https://portal.gdc.cancer.gov/
[2] https://www.cancerimagingarchive.net/
[3] https://www.naaccr.org/cina-research/
[4] https://www.frontiersin.org/journals/big-data/articles/10.3389/fdata.2025.1529848/full

mammographic images from 5,000 patients at King Abdullah University Hospital in Jordan. While the primary aim of the study was to enhance breast cancer classification using deep learning techniques, the dataset's construction and preprocessing also offer valuable insights into duplicate detection within clinical imaging data.

To improve image quality, the researchers applied several preprocessing methods, including high-boost filtering, contrast-limited adaptive histogram equalization (CLAHE), normalization, and data augmentation. While these techniques are essential for optimizing model performance, they can inadvertently introduce near-duplicate images if not carefully managed, potentially affecting the accuracy of training and evaluation processes.

The KAUH-BCMD dataset also includes metadata such as patient identifiers and diagnostic labels, which can be leveraged to link records and identify potential duplicates. Although duplicate detection was not the primary focus of the original study, the dataset's size and diversity make it a promising candidate for future research in this area. With the application of appropriate visualization and record linkage techniques, KAUH-BCMD could offer valuable insights into managing duplication challenges in cancer imaging data.

5) Breast Cancer Wisconsin (UCI)[5]*:* The Breast Cancer Wisconsin (Diagnostic) dataset, hosted by the UCI Machine Learning Repository, is a widely recognized benchmark in biomedical research. It comprises 569 samples derived from digitized fine needle aspirate (FNA) images of breast tumors, with each sample described by 30 numerical features related to the shape and texture of cell nuclei, such as radius, perimeter, area, and concavity. This rich feature set makes the dataset particularly valuable for developing and evaluating machine learning models in cancer diagnostics.

In terms of duplicate detection, the dataset is exceptionally clean and well-structured, featuring unique patient identifiers and no missing values. This reliability makes it an ideal starting point for testing deduplication algorithms or building classification models without concerns about data inconsistencies. However, when used in machine learning workflows, especially during cross-validation, care must be taken to avoid data leakage or overfitting. For instance, preprocessing steps like data augmentation or synthetic sample generation can unintentionally introduce near-duplicate entries, potentially skewing model performance.

Although the dataset does not contain explicit duplicates, its structured format and detailed metadata make it suitable for evaluating deduplication techniques, particularly those focused on feature-level similarity and record linkage. Researchers can simulate duplication scenarios by introducing controlled redundancy and applying visualization methods to detect and resolve them. Overall, the Breast Cancer Wisconsin dataset serves as a valuable resource for exploring deduplication strategies in small-scale, feature-rich clinical datasets, especially within the context of diagnostic imaging and classification.

While the datasets reviewed offer strong foundations for duplicate detection and visualization in cancer research, several limitations persist. The following section outlines key gaps in current approaches and proposes future directions to advance the field.

VI. Gaps and Future Research Directions

Current visualization approaches in clinical cancer datasets tend to emphasize general data quality issues or patterns of missing data. However, they often overlook specialized techniques for detecting duplicate records, especially in non-image data. Most existing tools rely on basic visual formats like tables, which can limit users' ability to interpret and effectively identify duplicates [3], [24], [26]. To address this gap, future research could focus on developing interactive visualization tools that are specifically designed to highlight duplicates within complex clinical datasets.

These tools should be capable of integrating temporal and multidimensional data, making it easier to spot and understand duplicate entries, ultimately improving data integrity and analytical precision. Moreover, current visualization systems rarely offer a comprehensive view of multiple data quality dimensions, such as completeness, accuracy, consistency, and temporal stability within duplicate detection workflows. When these aspects are only partially or implicitly represented, users struggle to assess the full impact of duplicates on overall data quality [2], [12], [21].

There is a clear opportunity to design integrated visualization frameworks that display these metrics alongside duplicate indicators, enabling a more holistic and intuitive evaluation of clinical data. Another limitation is the lack of support for visualizing how duplicates change or cluster over time. Temporal dynamics are especially important in clinical datasets, yet they are often underrepresented in current tools [20], [30]. This opens the door for innovative solutions like interactive timelines or change detection plots, combined with multidimensional views, to track duplicates longitudinally and provide deeper insights into data evolution.

VII. Conclusions

In conclusion, this review has examined current visualization techniques for detecting duplicates in clinical cancer datasets, particularly non-image data, and identified both promising developments and critical gaps. Despite progress in scalable algorithms and interactive analytics, most tools still rely on basic visual formats and fail to fully incorporate essential data quality dimensions such as completeness, accuracy, consistency, and temporal dynamics. There is a clear need for more intuitive and clinically relevant visualization frameworks that support complex, multidimensional data and allow users to explore duplicates interactively. Effective tools should be designed with users in mind, enabling human-in-the-loop workflows and facilitating the interpretation of anomalies over time. Many existing systems lack rigorous evaluation, standardized benchmarks, and long-term usability, which limits their practical impact. Future research should focus on building integrated platforms that combine robust deduplication

[5] https://archive.ics.uci.edu/dataset/14/breast+cancer

algorithms with user-friendly, scalable visual interfaces capable of supporting longitudinal tracking, cohort comparisons, and real-time data quality monitoring. Enhancing how duplicates are visualized is essential for improving data integrity, analytical accuracy, and ultimately patient outcomes. Bridging the gap between computational detection and visual understanding will empower users to not only identify duplicates but also take meaningful action, leading to more reliable cancer research and stronger foundations for data-driven healthcare innovation.

ACKNOWLEDGMENT

The authors would like to thank the Centre for Research and Innovation Management (CRIM), UTeM, and members of the IDEAL research group for all their support. This study is funded by the Malaysian Technical University Network (MTUN) Strategic Collaboration Research Grant (MTUN/2024/UTEM-FTMK/CRG/MS0008).

REFERENCES

[1] L. Waldron, M. Riester, M. Ramos, G. Parmigiani, and M. J. Birrer, "The doppelgänger effect: Hidden duplicates in databases of transcriptome profiles," Journal of the National Cancer Institute, vol. 108, no. 11, 2016, doi: 10.1093/JNCI/DJW146.

[2] H. Spengler, I. Gatz, F. Kohlmayer, K. A. Kuhn, and F. Prasser, "Improving Data Quality in Medical Research: A Monitoring Architecture for Clinical and Translational Data Warehouses," Computer-Based Medical Systems, pp. 415–420, 2020, doi: 10.1109/CBMS49503.2020.00085.

[3] D. Boehm et al., "Data visualization support for tumor boards and clinical oncology: Scoping review," JMIR Res Protoc., vol. 5:13:e5362, 2024, doi: 10.2196/53627.

[4] C. Turkay, F. Jeanquartier, A. Holzinger, and H. Hauser, "On computationally-enhanced visual analysis of heterogeneous data and its application in biomedical informatics," 2014.

[5] E. W. Prince and T. C. Hankinson, "for Clinical Decision Support," in Pacific Symposium on Biocomputing 2025, 2025, pp. 40–53, [Online]. Available: https://doi.org/10.1093/jamia/ocaf010.

[6] J. Wu, H. Wang, C. Ni, and K. Qian, "Interactive Data Visualization Techniques for Enhancing AI Decision Transparency in Healthcare Analytics: A Comparative Analysis," Applied and Computational Engineering, vol. 146, no. 1, pp. 175–186, 2025, doi: 10.54254/2755-2721/2025.tj22322.

[7] Q. Chen, J. Zobel, and K. Verspoor, "Duplicates, redundancies and inconsistencies in the primary nucleotide databases: A descriptive study," Database, vol. 2017, no. 1, pp. 1–16, 2017, doi: 10.1093/database/baw163.

[8] W. Digan, M. Wack, V. Looten, A. Neuraz, A. Burgun, and B. Rance, Evaluating the impact of text duplications on a corpus of more than 600,000 clinical narratives in a French hospital, vol. 264. IOS Press, 2019.

[9] J. Steinkamp, J. J. Kantrowitz, and S. Airan-Javia, "Prevalence and Sources of Duplicate Information in the Electronic Medical Record," JAMA Network Open, vol. 5, no. 9, pp. 1–11, 2022, doi: 10.1001/jamanetworkopen.2022.33348.

[10] B. Connolly et al., "A Statistical Approach for Visualizing the Quality of Multi-Hospital Data," Visible Language, vol. 48, no. 3, p. 68, 2014, [Online]. Available: https://www.questia.com/library/journal/1P3-3531587361/a-statistical-approach-for-visualizing-the-quality.

[11] D. G and S. Kumaresan, "A survey on duplicate record detection in real world data," International Conference on Advanced Computing, vol. 01, pp. 1–5, 2016, doi: 10.1109/ICACCS.2016.7586397.

[12] J. M. B. Josko and J. E. Ferreira, "Vis4DD: A visualization system that supports Data Quality Visual Assessment.," in 32th Brazilian Symposium on Databases, 2017, pp. 46–51, [Online]. Available: https://sbbd.org.br/2017/wp-content/uploads/sites/3/2017/10/proceedings-satellite-events-sbbd-2017.pdf.

[13] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate Record Detection: A Survey," Social Science Research Network, 2006.

[14] C. Forbes, H. Greenwood, M. Carter, and J. Clark, "Automation of duplicate record detection for systematic reviews: Deduplicator," Systematic Reviews, vol. 13, no. 1, 2024, doi: 10.1186/s13643-024-02619-9.

[15] M. Bilenko and R. J. Mooney, "Adaptive duplicate detection using learnable string similarity measures," in KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data minin, 2003, pp. 39–48, doi: 10.1145/956750.956759.

[16] G. N. Norén, R. Orre, A. Bate, and I. R. Edwards, "Duplicate detection in adverse drug reaction surveillance," Data Mining and Knowledge Discovery, vol. 14, no. 3, pp. 305–328, 2007, doi: 10.1007/S10618-006-0052-8.

[17] Y. Huang and F. Chiang, "Refining Duplicate Detection for Improved Data Quality," in TDDL, MDQual and Futurity Workshops at TPDL 2017, 2017, pp. 1–10, doi: https://doi.org/10.1145/3377878.

[18] M. I. Gabr, Y. Helmy, and D. S. Elzanfaly, "Tracking The Sensitivity of The Learning Models Toward Exact and Near Duplicates," International Journal of Advanced Computer Science and Applications, vol. 13, no. 12, 2022, doi: 10.14569/ijacsa.2022.0131240.

[19] O. Bieh-Zimmert, C. Koschtial, and C. Felden, "Representing multidimensional cancer registry data," in Proceedings of the 13th International Conference on Knowledge Management and Knowledge Technologies, 2013, pp. 1–4, doi: 10.1145/2494188.2494222.

[20] J. Scheer et al., "Visualization Techniques of Time-Oriented Data for the Comparison of Single Patients With Multiple Patients or Cohorts: Scoping Review," Journal of Medical Internet Research, vol. 24, no. 10, p. e38041, 2022, doi: 10.2196/38041.

[21] R. A. Ruddle, "Using Well-Known Techniques to Visualize Characteristics of Data Quality," Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: VISIGRAPP, VISIGRAPP, pp. 89–100, 2023, doi: 10.5220/0011664300003417.

[22] C. Turkay, R. S. Laramee, and A. Holzinger, "On the challenges and opportunities in visualization for machine learning and knowledge extraction: A research agenda," 2017.

[23] C. Schmidt et al., "Combining Visual Cleansing and Exploration for Clinical Data," in 2019 IEEE Workshop on Visual Analytics in Healthcare (VAHC), 2019, pp. 25–32, doi: 10.1109/VAHC47919.2019.8945034.

[24] [R. A. Ruddle, M. Adnan, and M. Hall, "Using set visualization techniques to investigate and explain patterns of missing values in electronic health records," medRxiv, 2022, doi: 10.1101/2022.05.13.22275041.

[25] H. M. Shakeel, S. Iram, H. Al-Aqrabi, T. Alsboui, and R. Hill, "A comprehensive state-of-the-art survey on data visualization tools: Research developments, challenges and future domain specific visualization framework," IEEE Access, vol. 10, pp. 96581–96601, doi: 10.1109/access.2022.3205115.

[26] S. J. Fernstad and J. Johansson, "To Explore What Isn't There -- Glyph-based Visualization for Analysis of Missing Values," IEEE Transactions on Visualization and Computer Graphics, vol. 28, no. 10, pp. 3513–3529, 2021, [Online]. Available: 10.1109/TVCG.2021.3065124.

[27] L. Hao, C. G. Healey, S. A. Bass, and H.-Y. Yu, "Visualizing Static Ensembles For Effective Shape and Data Comparison.," vol. 2016, no. 1, pp. 1–10, 2016, doi: 10.2352/ISSN.2470-1173.2016.1.VDA-509.

[28] R. Guo et al., "Comparative Visual Analytics for Assessing Medical Records with Sequence Embedding," Visual Informatics, vol. 4, no. 2, pp. 72–85, 2020, [Online]. Available: https://doi.org/10.1016/j.visinf.2020.04.001.

[29] C. N. Ta and C. Weng, "Detecting systemic data quality issues in electronic health records," Studies in Health Technology and Informatics, vol. 264, pp. 383–387, 2019, doi: 10.3233/SHTI190248.

[30] C. Sáez, P. P. Rodrigues, J. Gama, M. Robles, and J. M. García-Gómez, "Probabilistic change detection and visualization methods for the assessment of temporal stability in biomedical data quality," Data Mining and Knowledge Discovery, vol. 29, no. 4, pp. 950–975, 2015, doi: 10.1007/S10618-014-0378-6.

[31] A. Waldenburger, D. Nasseh, and J. Stausberg, "Detecting duplicates at hospital admission: Comparison of deterministic and probabilistic record linkage," Studies in Health Technology and Informatics, vol. 226, pp. 135–138, 2016, doi: 10.3233/978-1-61499-664-4-135.

[32] S. Shenoy, T.-T. Kuo, R. A. Gabriel, J. McAuley, and C.-N. Hsu, "Deduplication in a massive clinical note dataset.," To be published. Accessed: Sept. 25, 2025, [Online]. Available: https://arxiv.org/pdf/1704.05617v1.pdf.

[33] C. Wang and S. Karimi, "Parallel duplicate detection in adverse drug reaction databases with spark," in Proc. 19th International Conference on Extending Database Technology (EDBT), 2016, pp. 551–562, doi: 10.5441/002/edbt.2016.52.

[34] W. J. P. dos Santos et al., "A Scalable Parallel Deduplication Algorithm," Symposium on Computer Architecture and High Performance Computing, pp. 79–86, 2007, doi: 10.1109/SBAC-PAD.2007.32.

[35] J. Schmidt, "Scalable Comparative Visualization," 2016, doi: 10.2312/2631090.

[36] Q. Chen, J. Zobel, and K. Verspoor, "Benchmarks for measurement of duplicate detection methods in nucleotide databases," Database (Oxford), 2017, doi: 10.1093/DATABASE/BAW164.

[37] Q. Chen, J. Zobel, and K. Verspoor, "Duplicates, redundancies and inconsistencies in the primary nucleotide databases: A descriptive study," Database (Oxford), 2017, doi: 10.1093/DATABASE/BAW163.

[38] W. Andrzejewski, B. Bębel, P. Boinski, and R. Wrembel, "On tuning parameters guiding similarity computations in a data deduplication pipeline for customers records," Information Systems, 2023, doi: 10.1016/j.is.2023.102323.

[39] S. Sosvilla-Rivero, "Using set visualisation to find and explain patterns of missing values: a case study with NHS hospital episode statistics data," BMJ Open, vol. 12, no. 11, p. e064887, 2022, doi: 10.1136/bmjopen-2022-064887.

[40] A. M. Al-Mnayyis et al., "(KAUH-BCMD) dataset: advancing mammographic breast cancer classification with multi-fusion preprocessing and residual depth-wise network," Frontiers in Big Data, vol. 8, 2025, doi: 10.3389/fdata.2025.1529848.

Hybrid Real Time Facial Emotions Recognition on Autistic Individuals

Fatima Ezzahrae El Rhatassi[1], Btihal El Ghali[2], Najima Daoudi[3]
ITQAN Team-LYRICA Lab, Information Sciences School (ESI), Rabat, Morocco[1, 2]
SSLab-ENSIAS, Mohammed V University, Rabat, Morocco[3]

***Abstract*—Communication and social interaction issues are frequently linked to autism, which can have an impact on quality of life, work, and education. Opportunities to lessen these difficulties are presented by assistive technology, especially those that facilitate individualized and encouraging engagement. Facial expression recognition (FER) is essential to these systems, but current methods are still inadequate for autism-specific situations even though they achieve high accuracy on benchmark datasets like CK+. Because autistic people usually exhibit aberrant, subtle, or ambiguous facial expressions—which deviate from the common patterns used to train traditional FER models—this limitation occurs. In this work, we suggest a hybrid model that combines an LSTM network for temporal modeling of video sequences with three pretrained convolutional neural networks (EfficientNetB0, ResNet50, and MobileNetV2) for spatial feature extraction. Although the model performs well on CK+, its applicability to autism is still limited by the lack of relevant datasets and the use of artificial intelligence (AI)-generated videos rather than authentic recordings. The critical need for more comprehensive data and adaptive model designs tailored to autistic populations is highlighted by these findings.**

***Keywords*—Facial emotion recognition; video; frames; spatial and temporal features; pretrained models; LSTM; autistic individuals**

I. Introduction

Each of us needs to communicate since we belong to a community. Some people with autism spectrum disorders experience communication challenges. These individuals struggle to acquire language skills such as speaking and expressing their thoughts and emotions [1]. Emotion processing plays a critical role to help the autistic individuals face those challenges [2]. For this reason, we considered concentrating on the emotions of autistic people, which can be examined and identified from an image or live video when they communicate with a chatbot designed specifically for this population. While existing video-based FER models achieve good performance on standard datasets, they often fail to capture the atypical and subtle expressions of autistic individuals, due to muted intensity, ambiguous variations, and non-standard temporal dynamics. The principle behind this project is that we try to use a chatbot that provides to the autistic a personalized healthcare through conversation exchange. The chatbot will analyze the autistic person's words and attempt to identify his facial emotions and sentiments in order to help him feel better and avoid any urgent situations wherein he might need to notify parents, medical assistance, or other parties. Accordingly, the main objective of this study is to develop, evaluate, and adapt a hybrid FER model that combines multiple pretrained CNNs (EfficientNetB0, ResNet50, and MobileNetV2) with an LSTM for temporal modeling. The research is guided by the following question: Can such a multi-feature CNN–LSTM architecture improve the recognition of atypical autistic facial expressions in real-time video sequences? In this work, we limit the scope to facial expressions only, without including posture, gestures, or eye contact. While autistic people's emotions have been detected by picture analysis in earlier work [3], in this study, we try to uncover the challenges that arise while analyzing an autistic person's face in real time.

To maintain the logical structure, we will discuss the following areas: The related works that used video-based facial expression identification techniques rather than image-based ones will be shown in Section II. Our proposed work will be presented in Section III and the findings will be addressed. In Section IV, we discuss the usefulness and significance that this research has for individuals with autism. Section V presents the conclusion and future work.

II. Related Work

The majority of existing efforts have focused on the independent analysis of static images and their extension to videos, aiming to identify a person's emotion based on facial cues in those frames [4]. Consequently, research has shifted from relying solely on spatial feature extraction toward incorporating temporal feature modeling to capture the dynamic evolution of expressions. In parallel, a few initiatives have attempted to address real-time facial emotion recognition (FER) specifically for individuals with autism. For instance, Talaat et al. [22] developed a CNN and IoT-based framework, achieving up to 94.6% accuracy in real-time conditions while reducing latency through for computing. Similarly, Abou El-Magd et al. [23] proposed a hybrid kernel autoencoder–CNN model, which obtained 96.7% training accuracy and 95.4% validation accuracy on small autism-related datasets. Other efforts focused more on assistive systems: Lee and Wong's AEGIS [24], an augmented reality application, reported real-time emotion recognition with an average accuracy of 92%. However, such studies remain scarce and are often constrained by small or synthetic datasets, limited reporting of latency under real-world conditions, and the absence of large-scale evaluations on naturalistic autistic populations. Thus, while promising, current autism-focused FER research still lacks the robustness and generalizability of benchmark-oriented studies (e.g., CK+), which underscores the need for spatio-temporal

models optimized for real-time use and trained on representative autistic datasets. For example, Jad Haddad and his associates [5] think that temporal information is essential for tracking minor facial changes that occur when an emotion is expressed. They tried to build an efficient 3D-CNN for emotion recognition in videos, which takes as input a collection of sequential frames and analyzes them collectively, while maintaining the temporal aspect of a video sequence. The Optuna framework for effective hyper-parameter search was utilized to identify the best architecture for the 3D CNN model, which also contained the temporal hyperparameters. According to certain 2017 studies [6], the mouth and eye are the most crucial features for recognizing facial emotions. The eye and mouth areas are extracted using Temporal Gabor Features. Following normalization using the Z-score normalization procedure, they are encoded into binary pattern features, which are then concatenated to produce improved temporal features. Performance on the RML and CK datasets is significantly improved by this technique.

Some people use HappyER-DDF, a technique designed to identify pleasant emotions in videos [12]. In order to extract spatial-temporal data from sequential video frames, this system used a 3D-Inception-ResNet neural network. This method is novel by the fact that it incorporates a Long Short-Term Memory (LSTM) unit to record temporal dynamics, which is essential for precisely identifying emotions in videos.

Several CNN models, including VGG-Face, ResNet18, Densenet121, and VGG16, are used in other publications as [7] to handle facial images. A two-layer Bi-LSTM is then used to collect dynamic information. By combining each model's advantages, the fusion approaches seek to increase overall accuracy [8], [9], [10]. It can involve applying Principal Component Analysis (PCA) to a single emotional video in the two phases of preprocessing and feature extraction, combining other CNN architectures in the emotion classification phase [11], and analyzing emotion expressions using models such as VGG-FACE, LSTM (Long Short-Term Memory networks), and Xception with a statistic encoder (STAT) [9]. Generally the use of models or the idea of combining many models or architectures depend on the computational and time constraints in addition to the dataset used on training that impact automatically the results.

Facial expression recognition has advanced significantly in recent years, according to numerous research studies. For spatiotemporal feature extraction, some suggest merging deep learning and dynamic texture techniques, such as LSTM and VGG19 [13]. By incorporating a conventional Vision transformer (VIT), multi-View Complementary prompters (MCPs), and temporal-Modeling Adapters (TMAs) into the image model, others attempted to modify static image models for dynamic video-based emotion recognition [14]. In addition to face emotions, physiological information such as heart rate are presented on a video-based multimodal [15]. To racalibrate the channel-wise feature maps, a new framework is proposed by combining a 3D Xception network with Squeed-and-excitation modules. The HRV characteristics and iPPG signals were chosen as the two parameters that best illustrate the feasibility and promise of multimodal fusion.

There aren't many datasets utilized on this subject in general; we can include AFEW, CK+, and MMI, and they typically require a demand from their owner in order to be used. By focusing on temporal features and employing a variety of methods, including handcrafted descriptors, spatio-temporal aggregation, multimodal fusion, and pre-trained models, each paper advances video-based face emotion recognition. The various methods employed in various studies for facial expression identification, the dataset utilized, and the model's accuracy have been gathered in the next Table I.

TABLE I. PAPERS CONTRIBUTIONS AND TECHNIQUES TO ADVANCING VIDEO-BASED FACIAL EMOTION RECOGNITION

Objectives	Dataset	Techniques	Accuracy
Enhance emotion recognition in visual data by employing a deep neural network-based fusion model [9]	AFEW 2016, SAVEE, CEV data	VGG-FACE LSTM Xception	Accuracy of 54.83%
Recognizing happy emotions from videos using combined 3D visual information with deep learning techniques [6]	AM-FED+, AFEW, MELD	3D-Inception-ResNet	Accuracy of 95.97%
Recognize the Facial Emotion in Video Sequences Using Eye and Mouth Temporal Gabor Features [12]	RML and CK datasets	Gabor Wavelets, ensemble classifier, temporal feature encoding	The accuracy of Temporal feature extraction exceeds 90% for most of the emotions
Bi-modality Fusion for Emotion Recognition in the Wild [17]	AffectNet dataset, EmotiW	Video: VGG16,CNN,Bi-LSTM Audio: low level descriptor (LLDs)	Accuracy of 62.78%
EmotiEffNet and Temporal Convolutional Networks in Video-based Facial Expression Recognition and Action Unit Detection [13]	eNTERFACE05 database	VGG19, LSTM, HOG-HOF descriptors, Support Vector Machine (SVM)	Accuracy of 98%
From Static to Dynamic: Adapting Landmark-Aware Image Models for Facial Expression Recognition in Videos [14]	RAF-DB, AffectNet7/8 , and FERPlus, DFEW, FERV39K, and MAFW	Landmark-aware pre-trained models, temporal adapters	Accuracy varies from 25% to 93%. It depends on the emotion.
Video-Based Multimodal Spontaneous Emotion Recognition Using Facial Expressions and Physiological Signals [15]	BP4D+	Multimodal fusion of facial and physiological features	Accuracy of 70%
Real-Time Facial Emotion Recognition Using Deep Learning Models [16]	BAUM-1s, eNTERFACE05	3D Convolutional Neural Networks, spatial–temporal feature aggregation	Around 45% of accuracy

III. Proposed Work: The Fusion of Three Models

The identification of facial expressions from images or videos has been the subject of numerous studies; nonetheless, the fundamental framework is essentially composed of the same components as shown in Fig. 1.

At the data collection stage, the focus is on gathering facial expression datasets from videos. Several datasets are publicly available, and among them, we chose CK+. This choice was motivated not only by the fact that CK+ has been pretrained on a large amount of data, but also because it was the only dataset provider who responded positively to our request and granted us access. In addition, CK+ presents significant advantages compared to static image datasets, as it contains complete temporal sequences:

- Neutral starting point: Each sequence begins with a neutral facial expression.
- Progressive evolution: The dataset captures the gradual development of the expression until its peak intensity.
- Detailed annotations: CK+ provides more than just a global label for each video.
  - Sequence-level labels: Each video sequence is labeled with a single primary emotion, ensuring clarity for training.
  - Apex frame annotations: The exact frame where the expression reaches its peak intensity is explicitly identified.

These characteristics make CK+ particularly suitable for our study, especially when combined with a fusion of pretrained models—EfficientNetB0, ResNet50, and MobileNetV2—used for the training stage.

Generally, the preprocessing stage involves identifying faces in every video frame utilizing deep learning-based or face identification algorithms [17]. We included cropping and aligning faces to guarantee uniformity in facial areas between frames and adjusting the photos for changes in scale, position, and lighting. It is essential to gather temporal and spatial feature [18, 19] analysis in order to build a performant model that considers the relationship between the frames in order to recognize face emotions from a video.

The architecture that we are proposing is illustrated below in Fig. 2.

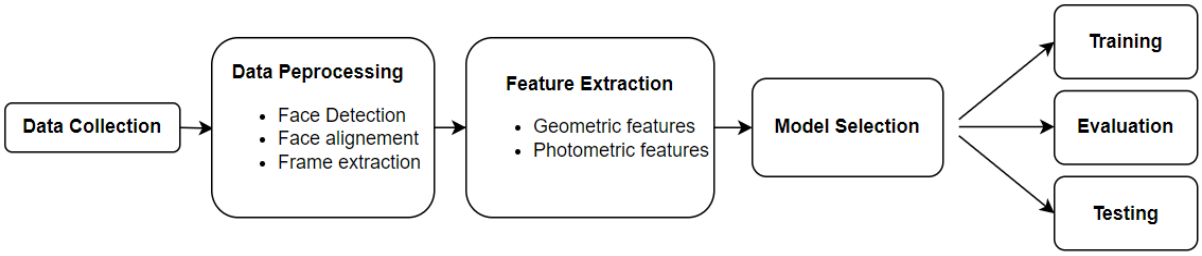


Fig. 1. The process of facial emotion recognition.

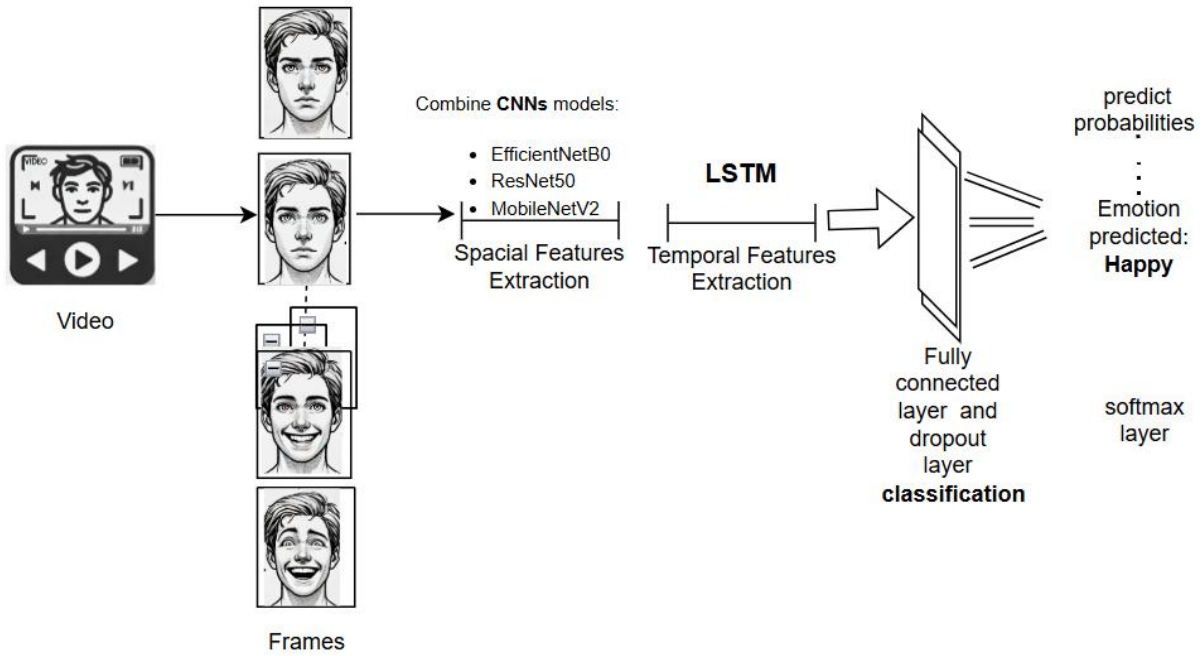


Fig. 2. The model of facial emotion recognition by combining CNNs models and LSTM for features extraction.

We will be using the combination of CNN (Convolutional Neural Networks) for spatial feature extraction and RNNs (Recurrent Neural Networks) for temporal dependencies by the use of LSTM (Long Short-Term Memory).

In order for our model to learn from the CK+ dataset, which is made up of folders with sequential frames but no individual image labels, the first step is to label these sequential frames. Each folder in the CK+ dataset represents an emotion sequence, with the final frames showing the peak of an emotion (such as happy, angry, or surprised) and the initial few frames usually showing a neutral face. In order to read sequences (video frames) and the labels that go with them, we first load the CK+ dataset (pictures and labels) into memory for the training and validation phases. We then cycle through image folders. Then, for multi-class classification, we use to_categorical to transform labels into one-hot encoding.

For efficiency, we preprocess the data before storing it in npy files and utilizing train_test_split (80%-20% split) to divide the dataset into training and validation sets.

We create a Multi-Feature Base Model by creating a base feature extractor model using the pre-trained networks EfficientNetB0, ResNet50, and MobileNetV2. We extract features from three pre-trained models for each frame and use TimeDistributed to apply the same feature extractor across all frames in a sequence and finally combine features from all three models using Concatenate. After that we add a LSTM layer to model temporal dependencies in the feature sequences and for the classification we add fully connected and dropout layers and final output layers uses softmax activation to predict probabilities for eight classes (anger, contempt, disgust, fear, happiness, neutrality, sadness and surprise).

The choice of this method is motivated by several factors:

- Complementary strengths of CNN and LSTM: CNNs excel at capturing spatial details from facial regions, while LSTMs are effective in modeling the temporal evolution of expressions across frames.
- Robustness through model fusion: By combining EfficientNetB0, ResNet50, and MobileNetV2, we leverage the diversity of their learned representations, improving generalization compared to using a single model.
- Efficient handling of temporal sequences: The use of TimeDistributed allows frame-by-frame processing while maintaining temporal coherence, which is crucial for datasets like CK+ that provide full expression sequences.
- Scalability and flexibility: The architecture can be easily extended to incorporate additional pretrained models or adapted to other datasets with minimal adjustments.
- Improved classification accuracy: The multi-feature extraction, combined with LSTM's temporal modeling, enhances the system's ability to capture subtle expression changes, which are often missed in static-image approaches.

Following these processes, it's time to assess the model, examine how well it performs across a range of emotions, and determine its limitations.

IV. Experimental Results

A. Applying the Model to CK+ Dataset

Fig. 3 illustrates the training and validation accuracy and loss curves over the course of the training process, providing insights into the model's performance and generalization.

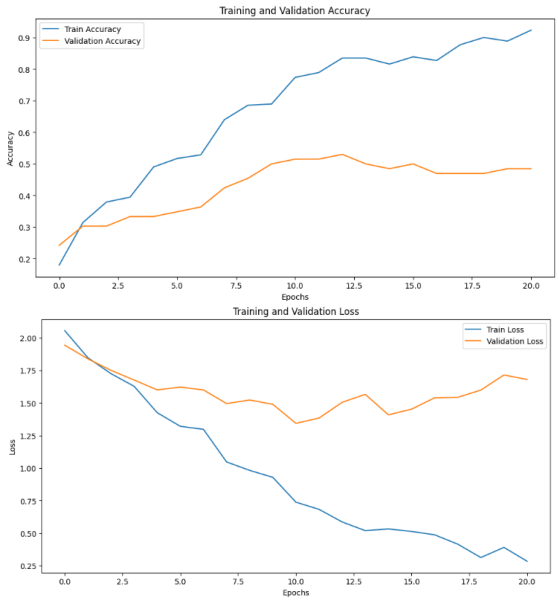


Fig. 3. The model's training and validation accuracy/loss.

1) Accuracy and Loss: Between Epochs 1 and 21, the training accuracy increases progressively from 17% to 94.5%. The model is successfully learning from the training data when the training loss steadily drops.

2) Validation Accuracy and Loss: In Epoch 13, the validation accuracy peaked at 53.03%, having begun at 24.2%. After that, though, it starts to plateau and even slightly declines (for example, 48.48% in Epoch 21). After initially declining, the validity loss starts to change at about Epoch 10. This suggests that the model has trouble generalizing outside of the training set.

3) The model's correct predictions: Here in Fig. 4 are some of correctly classified images along with their corresponding true labels and predicted labels that visually inspects how well the model has performed on correctly predicted samples.

4) The model's false predictions: The model successfully learns from the training data but struggles to generalize, as evidenced by the images below in validation data in Fig. 5.



Fig. 4. A sample of correctly predicted pictures.



Fig. 5. A sample of incorrect predicted pictures.

In our experiments, we implemented a hybrid architecture combining CNNs and LSTMs. Spatial features were extracted using three pretrained backbones (EfficientNetB0, ResNet50, and MobileNetV2), whose outputs were concatenated to form a multi-feature representation. Temporal dependencies across video sequences were then modeled using LSTM layers, followed by fully connected layers for classification.

While the training accuracy reached 94.5 % after 21 epochs, the validation accuracy plateaued around 53 %. This performance is significantly below the state-of-the-art that worked with the same dataset as [20,21], where recent works report accuracies above 95–98 % on CK+ using attention-based CNNs or spatio-temporal models such as FAN or CAGNet. We attribute this gap mainly to overfitting due to the high model complexity relative to the dataset size, and the absence of attention mechanisms to focus on the most informative frames or facial regions.

Nevertheless, our architecture demonstrates the feasibility of combining multiple pretrained models with temporal modeling, aligning with current research trends. Future work will focus on reducing overfitting through data augmentation, regularization, and possibly integrating attention modules to improve generalization and bring performance closer to the reported state-of-the-art.

B. Applying the Model to Videos of Autistic Children

In our last research [2], we worked with images of autistic children available online, gathered by Fatima Talaat at https://www.kaggle.com/datasets/fatmamtalaat/autistic-children-emotions-dr-fatma-m-talaat. Since we face a huge problem related to data and privacy, especially when it comes to people with autism, it is so hard to get acceptance to use the data even if it will not be shared. The lack of available datasets that can respond to this need in order to validate our proposed approach made us think about using the artificial intelligence tools to make videos only from the images that we worked with previously. The tool used in our case is https://www.HeyGen.com/ which is an AI-driven platform that enables users to create AI-generated videos quickly and efficiently. By selecting an image of the autistic child and expressing the setting and emotion we wanted it to convey, we attempted to produce a customized video content. We then applied our fusion model to it.

The steps to predict the final emotion are as follows in Fig. 6.

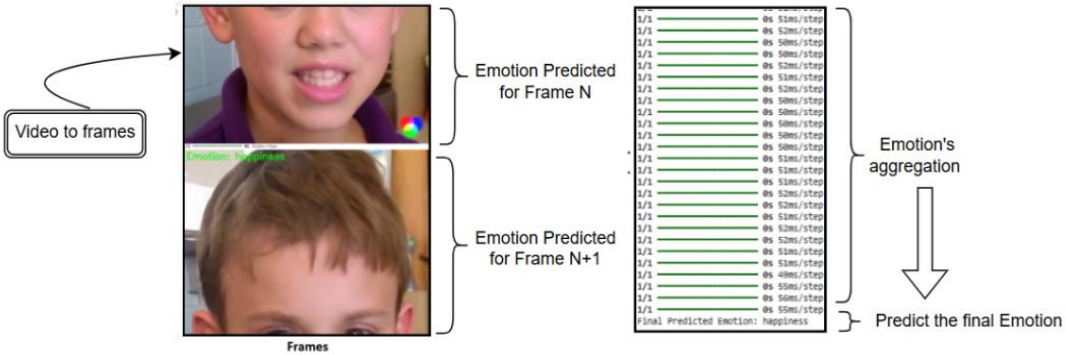


Fig. 6. The schema of splitting frames and predicting the final emotion.

This schema describes the pipeline for video-based emotion recognition, focusing on how frames are processed and predictions are made for sequences of frames in our model application.

1) Video processing by breaking the video containing the autistic into frames: A video is essentially a sequence of static images (frames) displayed over time. The first step is to

extract these individual frames from the video and to preprocess only frames with faces.

Each frame is treated as an image, enabling the model to process it independently or in groups. Frame-by-Frame Processing: This method ensures that every segment of the video is examined chronologically by processing it frame by frame. Since emotions in a video frequently change over time, this is essential for preserving the temporal context.

2) Sequence-based emotion prediction by processing sequences of frames: The pipeline aggregates frames into sequences of a fixed size (e.g. time_steps = 10) rather than predicting emotions for a single frame separately. This method captures temporal dependencies, which are crucial for comprehending dynamic emotions because of the interaction between successive frames.

3) Emotion's prediction where the model predicts the corresponding emotion (e.g., happy, sad, angry...) for every set of frames. These sequences are handled by temporal models, LSTMs which can learn from patterns and changes in the frames over time.

4) Aggregation of predictions by applying dominant emotion inference: The model combines the predictions made after analyzing every frame sequence to identify the main emotion expressed in the video. For example, if the majority of the scenes are categorized as "happy," then "happiness" may be the general feeling conveyed by the video. Voting, averaging probability, or choosing the most frequent forecast are some examples of aggregation techniques.

5) Result of applying the model on videos containing autistic individuals: The model performs well overall but struggles with many false predictions. These can be related to some autistic children who exhibit atypical or subtle facial expressions of emotions as muted smiles or ambiguous anger or fear expressions, which differ from what we had in the training data in terms of the expressed facial emotions. Also, the generative AI videos may not perfectly capture real-world variations or traits in autistic expressions, which can help create bias.

Our model needs some improvements in having more diverse, real-world dataset of autistic expressions and architecture's adjustment to focus on subtle facial regions as mouth corners, eyebrows.

Several challenges arise when applying FER in real time for individuals on the autism spectrum:

- Subtlety and variability of expressions: Autistic individuals may display emotions in less pronounced or non-standardized ways, making it difficult for models trained on typical datasets to recognize them accurately.
- Reduced or inconsistent eye contact: Since many FER systems rely heavily on eye-region cues, this can lead to incomplete or misleading feature extraction.
- Contextual dependence: Some expressions may not be directly tied to a single emotion but are influenced by the situational context, requiring the system to interpret beyond facial features.
- Sensory and behavioral differences: Atypical movements, self-stimulatory behaviors (stimming), or avoidance of camera focus may interfere with real-time face detection and tracking.
- Latency and real-time constraints: Achieving both high accuracy and low processing delay is particularly challenging when expressions are subtle and rapidly changing.
- Dataset limitations: The scarcity of large, diverse, and ethically collected datasets of autistic individuals further restricts the model's ability to generalize.

To address these issues, our model requires improvements in two main directions: 1) access to more diverse and representative real-world datasets of autistic expressions, and 2) architectural adjustments that give greater attention to subtle facial regions such as mouth corners, eyebrow movements, and micro-expressions.

V. Conclusion and Future Work

For the identification of emotions in videos, our approach is based on an architecture that allows the combination of several pre-trained models. We chose to merge the three pre-trained CNNs (EfficientNetB0, ResNet50, and MobileNetV2) with a temporal model based on LSTM in order to extract both spatial features (using CNN) and temporal relationships (using LSTM) from the video sequence. On the CK+ dataset, our model showed strong learning capability, with training accuracy reaching 94.5%. However, the validation accuracy plateaued at 53.03%, which highlights the model's limited ability to generalize beyond the training data and confirms the difficulty of applying FER to real-world autistic expressions. Because we intend to interact with a group of people with autism, and since their facial expressions in a variety of scenarios are unique, recognizing emotion is difficult even in the absence of an AI tool. We have currently attempted to use the AI technologies to create films in order to apply our model to them using datasets of public images. In order to improve the model's performance, we will try to concentrate on this community in our upcoming works, collect a database of data for training, and try to pay attention to the small details that make a difference in the autistic face with medical assistance. We might additionally modify the architecture to make it more consistent and performant to be applied to this kind of population or integrate contextual cues (e.g. body language, vocal tones) to reduce reliance on facial data alone. To achieve even more substantial findings that are aligned with the specificity of these populations and accepted by the medical party, many parties need to work together and concentrate on understanding facial features and emotions.

References

[1] Frith, Uta. (1989). A new look at language and communication in autism. The British journal of disorders of communication. 24. 123-50. 10.3109/13682828909011952.

[2] F. E. E. Rhatassi, B. E. Ghali, and N. Daoudi, “Deep Learning Approaches for Recognizing Facial Emotions on Autistic Patients”

[3] Development of video-based emotion recognition using deep learning with Google Colab published in TELKOMNIKA Vol. 18 No. 5, October 2020.

[4] Jad Haddad, Olivier Lezoray, and Philippe Hamel. 2020. 3D-CNN for Facial Emotion Recognition in Videos. In Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II. Springer-Verlag, Berlin, Heidelberg, 298–309. https://doi.org/10.1007/978-3-030-64559-5_23

[5] Rani, P.I., Muneeswaran, K. Recognize the facial emotion in video sequences using eye and mouth temporal Gabor features. Multimed Tools Appl 76, 10017–10040 (2017). https://doi.org/10.1007/s11042-016-3592-y

[6] Sunan Li, Wenming Zheng, Yuan Zong, Cheng Lu, Chuangao Tang, Xingxun Jiang, Jiateng Liu, and Wanchuang Xia. 2019. Bi-modality Fusion for Emotion Recognition in the Wild. In 2019 International Conference on Multimodal Interaction (ICMI '19). Association for Computing Machinery, New York, NY, USA, 589–594. https://doi.org/10.1145/3340555.3355719

[7] Muhammad, F.; Hussain, M.; Aboalsamh, H. A Bimodal Emotion Recognition Approach through the Fusion of Electroencephalography and Facial Sequences. Diagnostics 2023, 13, 977. https://doi.org/10.3390/diagnostics13050977

[8] Do, LN., Yang, HJ., Nguyen, HD. et al. Deep neural network-based fusion model for emotion recognition using visual data. J Supercomput 77, 10773–10790 (2021). https://doi.org/10.1007/s11227-021-03690-y

[9] C. Gan, J. Yao, S. Ma, Z. Zhang, and L. Zhu, “The deep spatiotemporal network with dual-flow fusion for video-oriented facial expression recognition,” Digital Communications and Networks, vol. 9, no. 6, pp. 1441–1447, Dec. 2023, doi: 10.1016/j.dcan.2022.07.009.

[10] Hajarolasvadi, Noushin & Demirel, Hasan. (2020). Deep Facial Emotion Recognition in Video Using Eigenframes. Image Processing, IET. 10.1049/iet-ipr.2019.1566.

[11] N. Samadiani, G. Huang, Y. Hu and X. Li, "Happy Emotion Recognition From Unconstrained Videos Using 3D Hybrid Deep Features," in IEEE Access, vol. 9, pp. 35524-35538, 2021, doi: 10.1109/ACCESS.2021.3061744.

[12] Chouhayebi, H.; Mahraz, M.A.; Riffi, J.; Tairi, H.; Alioua, N. Human Emotion Recognition Based on Spatio-Temporal Facial Features Using HOG-HOF and VGG-LSTM. Computers 2024, 13, 101. https://doi.org/10.3390/computers13040101

[13] Chen, Y., Li, J., Shan, S., Wang, M., & Hong, R. (2024). From static to dynamic: Adapting Landmark-Aware image models for facial expression recognition in videos. IEEE Transactions on Affective Computing, 1–15. https://doi.org/10.1109/taffc.2024.3453443

[14] Y. Ouzar, F. Bousefsaf, D. Djeldjli and C. Maaoui, "Video-based multimodal spontaneous emotion recognition using facial expressions and physiological signals," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), New Orleans, LA, USA, 2022, pp. 2459-2468, doi: 10.1109/CVPRW56347.2022.00275.

[15] Naveen N.C., Sai Smaran K.S., Shamitha A.S (2024). Real Time Facial Emotion Recognition using Deep Learning Models,"International Journal of Computer Applications (0975 – 8887) Volume 186 – No.29

[16] Li, S., Zheng, W., Zong, Y., Lu, C., Tang, C., Jiang, X., Liu, J., & Xia, W. (2019). Bi-modality Fusion for Emotion Recognition in the Wild. 2019 International Conference on Multimodal Interaction.

[17] N. T. Singh, S. Rana, S. Kumari and Ritu, "Facial Emotion Detection Using Haar Cascade and CNN Algorithm," 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT), Kollam, India, 2023, pp. 931-935, doi: 10.1109/ICCPCT58313.2023.10245125.

[18] K. Zhang, Y. Huang, Y. Du and L. Wang, "Facial Expression Recognition Based on Deep Evolutional Spatial-Temporal Networks," in IEEE Transactions on Image Processing, vol. 26, no. 9, pp. 4193-4203, Sept. 2017, doi: 10.1109/TIP.2017.2689999.

[19] Z. Xia, X. Hong, X. Gao, X. Feng and G. Zhao, "Spatiotemporal Recurrent Convolutional Networks for Recognizing Spontaneous Micro-Expressions," in IEEE Transactions on Multimedia, vol. 22, no. 3, pp. 626-640, March 2020, doi: 10.1109/TMM.2019.2931351.

[20] Fan, Y., Lam, J. C. K., Li, V. O. K., & Kot, A. C. (2020). Frame attention networks for facial expression recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 44*(9), 6096–6108. https://doi.org/10.1109/TPAMI.2020.2974829

[21] Zhang, X., Wang, Y., Zhang, C., & Liu, C. (2021). Hybrid attention cascade network for facial expression recognition. *Sensors, 21*(6), 2003. https://doi.org/10.3390/s21062003

[22] Talaat, F. M., Abou El-Magd, H., & El-Hosseini, M. A. (2023). Real-time facial emotion recognition system among children with autism based on deep learning and IoT. Neural Computing and Applications, 35(29), 21023–21038.

[23] Abou El-Magd, H., El-Hosseini, M. A., & Talaat, F. M. (2024). Real-time facial emotion recognition model based on kernel autoencoder and convolutional neural network for autism children. *Soft Computing, 28*(6), 14873–14891

[24] Lee, J. H., & Wong, K. W. (2020). AEGIS: A real-time multimodal augmented reality computer vision based system to help individuals with autism spectrum disorders recognize emotions. arXiv.

Integrating YOLOv8 and IoT in a Computer Vision System for Child Detection in Smart Cities

Modhawi Alotaibi*, Atheer Alruwaythi, Sara Alenazi, Maisaa Alsaedi
College of Computer Science and Engineering, Taibah University, Madinah, Saudi Arabia

***Abstract*—In an era marked by technological advancements aimed at establishing smart cities, technology increasingly focuses on enhancing aspects related to crowd management. The widespread deployment of CCTV systems, combined with the integration of computer vision, has enabled accurate insights into crowd density estimation. Our research highlights the potential benefits of child detection across various domains that serve governments and business decision-making. Leveraging Internet of Things (IoT) devices to collect real-time data and employing artificial intelligence (AI) based on deep learning through computer vision is powerful in such domains. In this paper, we propose an IoT architecture that facilitates intelligence and decision-making in two phases: 1) a deep learning model with object detection and image segmentation capabilities using YOLOv8, and 2) a tracking/counting algorithm for estimating child density based on DeepSORT. Our implementation efficiently identified and classified children in extracted images with an accuracy rate of up to 98%. Also, our model outperformed the other two solutions proposed by previous studies in terms of mAP@50, Precision, and Recall metrics. The results provide valuable insights for businesses aiming to refine site selection and guide governments in improving urban planning and safety, thereby fostering sustainable and intelligent urban development.**

***Keywords*—Computer vision; Internet of Things; deep learning; YOLOv8; DeepSORT**

I. Introduction

With the widespread deployment of Closed-circuit Television (CCTV) surveillance systems, computer vision plays a vital role in various applications based on crowd and flow density analysis. In an era characterized by technological shifts to smart cities, there is a focus on developing and enhancing aspects related to crowd management. One interesting application is the use of crowd analysis for child detection. The research field of child detection can be beneficial in various domains, such as advertising and business site selection. Governments can also gain significant advantages from child detection in domains such as public safety and law enforcement, as well as from collecting demographic information accurately for enhanced urban planning. Child detection can play a crucial role in addressing critical safety issues, such as supporting efforts to locate lost children [1, 2] and monitoring traffic to detect front-seat child occupancy in vehicles [3]. Additionally, governments can deploy child detection for smart urban planning to optimize site selection for various reasons [4], including areas targeted towards children (e.g. child centers, schools, and parks).

Despite the growing interest in crowd analysis, there remains a notable research gap in the accurate detection and tracking of children within dynamic urban environments using integrated IoT and AI technologies. Existing studies often focus on general crowd density, leaving child-specific analysis underexplored. This paper aims to bridge that gap by proposing a specialized framework for child detection and density estimation.

Distinguishing children from adults in digital images has become increasingly important for various applications, such as adjusting electronic device controls and tailoring advertisements [5], as well as informing business decisions. In today's competitive business landscape, selecting the right location is a make-or-break decision for new ventures. Businesses can utilize crowd and flow density analysis as a strategic method to gain valuable insights by accurately estimating foot traffic patterns and behaviors in a given area. Consequently, entrepreneurs can determine the optimal area for their target customers (including children), ultimately leading to enhanced commercial outcomes.

Boosting economic growth and safety through crowd data analysis to identify optimal areas for targeted audiences or enhancing the human experience lays the foundation for sustained growth and long-term success [6]. Emerging technologies like IoT and AI can play vital roles in such decision-making processes. To demonstrate the impact of crowd analysis, specifically child detection, on enhancing smart urban planning, we integrate IoT, computer vision, and deep learning technologies, focusing on areas where children are the primary beneficiaries.

IoT has become pervasive and intuitive in the technical, economic, and social fields of smart cities. It works by incorporating internet connectivity with powerful data analysis capabilities [7]. IoT deploys sensing devices to collect data and then transforms raw data into actionable insights. This process requires sensors that generate data to be collected, analyzed, and visualized to obtain useful information [8]. The data is then interpreted to produce decisions in a deliverable form [9]. The massive volume of data produced by IoT devices highlights the importance of AI, which efficiently manages and processes this vast data [10].

AI is a field of computer science that studies technologies allowing computers to intelligently simulate human cognitive functions [11]. AI encompasses various components, including deep learning, a subset of machine learning (ML) that enables machines to learn by analyzing inputs using neural networks inspired by the structure of the human brain's neurons. These

*Corresponding Author.

networks consist of layers of interconnected neurons arranged in a specific pattern and are specialized for processing a grid of values. Cognitive computing emulates human cognitive abilities, including sensing, perception, learning, decision-making, and action When integrated with IoT, cognitive computing can enhance surveillance systems to enable automated and intelligent crowd management, which understands context better and responds efficiently [12].

In the context of crowd analysis, leveraging IoT and AI, primarily through computer vision, enables accurate crowd detection and analysis. Computer vision techniques help analyze surveillance videos to extract valuable insights. It utilizes cameras, data, and algorithms to process information quickly and efficiently [13]. Computer vision techniques provide solutions to current challenges in crowd analysis, such as tracking accuracy and density estimation [14]. Our research discusses how these techniques detect children within crowd videos, which can benefit various applications across multiple domains.

Our main contribution in this paper is leveraging AI tools, IoT, and computer vision to improve urban site selection decisions for child-targeted areas. By adopting IoT devices such as cameras, we collect vast amounts of real-time data from various locations and process it with advanced AI techniques to accurately interpret crowd characteristics and behaviors. Computer vision plays a crucial role in this process, providing precise visual data analysis and applying tracking algorithms that detect, track, and estimate child density in specific areas. We propose leveraging these technological advancements to make smarter decisions, emphasizing the significant impact of our study on optimizing urban planning for sites targeted toward children. This work has significant implications for governments, urban planners, and businesses, enabling data-driven policies, enhanced safety measures, and improved infrastructure planning.

In addition to the introduction in Section I, this paper offers five more sections. Section II reviews related work on deploying IoT and intelligent methods, such as computer vision and deep learning in the crowd analysis domain. Section III describes the proposed IoT architecture. In Section IV, we discuss in detail the methodology. We present the results of our proposal in Section V. Finally, Section VI concludes the paper.

II. Related Work

In this section, we review other studies relevant to our research. The domain of crowd analysis has been addressed from various angles using different enabling tools. One primary tool is IoT. Solmaz et al. [15] presented a study on managing crowd mobility in smart cities using IoT technologies. They introduced the federated interoperable semantic IoT (FIESTA-IoT) platform, which enables information sharing among stakeholders for efficient crowd management. They addressed challenges such as the scalability of connected systems, information transparency between different systems, data federation, and information privacy. In another study, Varghese et al. [12] discussed the application of cognitive computing for smart crowd management in a surveillance system. The authors showcased two case studies: smart crowd management during Hajj (an Islamic mega event) and smart crowd management during the Tour de France (an annual bicycle race). Their IoT-based intelligent crowd-management system utilized mobile sensor data along with video data for efficient monitoring. They employed various AI algorithms, such as deep learning, to analyze IoT big data for real-time insights. In the context of mega gatherings, crowd data should be collected and utilized for better management. The authors [16] proposed an intelligent IoT approach for crowd management during Hajj to avoid congestion. Their solution was built based on the characteristics of Mina (a holy area in Makkah). In this proposed approach, they constructed the IoT sensing layer using Radio Frequency Identification (RFID) cards and surveillance cameras. Then, based on calculated risk factors, a routing algorithm guides pilgrims through safe paths. However, this proposal lacks the use of AI tools, which could elevate the solution and its outcomes. Mohd et al. [8] discussed current IoT research technologies and applications for serving Hajj and Umrah pilgrimages based on crowd analysis. They proposed practical suggestions that could employ technologies to facilitate services provided to pilgrims, Umrah performers, visitors, and service providers. The first service they suggested was a smart-screen device to prevent cases of lost pilgrims and Umrah performers. The second was a monitoring system with automated alerts to detect and respond quickly to unusual events. Third, they proposed a smart parking system using devices at each parking lot to help people locate their vehicles via plate numbers and interactive maps.

A second enabling tool is computer vision in conjunction with other technologies. Li et al. [17] examined crowd density in a tourist area using video management dynamic information analysis. They proposed a counting model that could detect crowd density through cameras. Additionally, they developed inverse-scale perception modules to support the extraction of multi-scale information. They used their model to generate a distribution-density map, counting the population based on convolutional neural networks (CNN). The authors of [18] proposed a framework that integrates CCTV video footage and spatial information, using deep learning models, geometric transformations, and tracking algorithms to extract data on crowd congestion. This approach—using deep learning techniques to accurately analyze and process visual information—allows real-time monitoring and prediction of crowd congestion, assisting urban planners and infrastructure operators in managing crowd congestion and improving public safety. Besides deploying crowd analysis for detecting abnormalities, it can also be used for flow prediction. Interestingly, Zhou et al. [19] focused on crowd-flow prediction for planned transportation sites, a critical challenge for urban planners and administrators. They offered a data-driven technique for predicting future population flows for newly planned sites.

In the domain of child safety, numerous studies have focused on techniques for child detection in images [2,3,20]. For instance, Basaran et al. [2] proposed the ChildSafe system, a solution for classifying individuals as children or adults using 3D depth cameras, aimed at enhancing child safety applications. The methodology involved collecting data from 193 participants, including 150 children and 43 adults aged 13 to 50, using Microsoft Kinect sensors. Participants performed

eight predefined actions, and the system extracted body lengths, facial metrics, and relative ratios to mitigate errors from absolute measurements. A bin-boundary-based classifier was applied to process these features, mapping them into discrete bins and calculating weights based on the likelihood of feature occurrences. The system achieved a high classification accuracy of up to 97% with low error rates. However, the solution faces challenges, particularly its reliance on controlled settings with predefined actions, which limits its effectiveness in real-world environments with dynamic lighting and unscripted behaviors. Balci et al. [3] presented a study on an automated system for detecting child occupancy in the front seat of vehicles on roadways, addressing a critical safety issue in traffic enforcement. They employed deep learning techniques to analyze images captured from an overhead gantry, achieving an overall accuracy of 90%. The methodology includes various approaches for face detection and passenger classification.

In other studies, targeting social security and crime prevention applications, the authors of [21,22] focused on classifying pedestrians as children or adults using biometric features from video data. Both approaches used the head-to-body length ratio for classification but differed in methodology, accuracy, and how they addressed challenges. In [21], a method for classifying pedestrians as children or adults using biometric features derived from long-distance video data was proposed. The study employed Haar-like features with Adaboost for full-body detection and utilized static thresholds for head-to-body ratio-based classification. The method achieved 74.7% accuracy for adults and 68.1% for children. However, challenges arose from inconsistent head detection, especially when head orientations deviated from the frontal view, and the reliance on static thresholds limited its adaptability in dynamic scenarios. Similarly, the authors of [22] proposed a framework that utilized Haar-like features with Adaboost for full-body detection and Local Binary Patterns (LBP) for head detection. This study introduced a moving average algorithm to dynamically adjust classification thresholds, addressing variations in pedestrian positions. The approach achieved 100% accuracy for children and 64.5% for adults, with lower accuracy for adults attributed to deviations in body proportions. On the other hand, [23] presented a method for detecting underage individuals using facial image classification. It utilized a dataset categorized into three age groups: children, adults, and the elderly. The methodology employs two main feature extraction techniques—Histogram of Oriented Gradients (HOG) and Haar features—to analyze facial characteristics, including wrinkles, geometric features, and facial ratios. The implementation uses the Viola-Jones algorithm for face detection, processing real-time images at two frames per second, making it suitable for various real-world applications such as online content restriction and age-age-restricted purchases.

One major benefit of crowd analysis is the ability to detect and track objects. In [24], the authors introduced DeepSORT, a tracking system that uses a neural network to enhance object tracking by combining motion and appearance information. DeepSORT aims to improve tracking accuracy by reducing identity switches and prioritizing frequently seen objects. DeepSORT achieves excellent results in tracking multiple objects and performs well across various scenarios, including our research focus on smart urban site selection for children's areas.

Selecting an appropriate site for a particular purpose is one of the many applications of crowd and flow analysis [25–28]. The purposes vary to serve different sectors, such as education, transportation, agriculture, entertainment, business, and health. Focusing on the concept of site selection from a business perspective, many authors strongly connect location selection to business success [29–31]. In [29], the author emphasized that investors must evaluate new and existing locations to determine the optimal sites for starting a business. Akpan et al. [30] studied the impact of retail outlet location and accessibility on business performance. They gathered data from businesses through a structured survey of 90 of 120 retail outlets considering functioning in an area. They analyzed the collected data using two statistical techniques: Structural Equation Modelling (SEM) and Relative Importance Index (RII). SEM is used to study the connection between retail location and profitability, while RII ranks the elements according to each factor's value in relation to other factors impacting company performance. In another study, Yakovlev et al. [31] examined the business-site-selection problem, explicitly addressing the maximum coverage location problem (MCLP). The MCLP is an optimization challenge that assists businesses in identifying the most suitable locations for their facilities to maximize service coverage. The study focused on the continuous formulation of the MCLP, which considers the arbitrary spatial shape of the demand zone (where services are needed) and the service areas (where facilities will be placed). Due to limitations in processing spatial and geometric data for the demand area, the demand zone is represented as a set of points. The study involved developing mathematical models and approaches to solve the planar MCLP using modern computer technologies. The authors emphasized the importance of considering the size and shape of the demand zone and service areas in the decision-making process for optimal facility placement.

The reviewed literature highlights substantial advancements in crowd management, child detection, and site selection through IoT, computer vision, and deep learning. However, several limitations persist. Many crowd management systems lack robust AI integration, reducing their adaptability and predictive capabilities in dynamic environments. Child detection methods often depend on controlled settings or static thresholds, which limit their effectiveness in real-world scenarios characterized by variable lighting, movement, and behavior. Tracking algorithms like DeepSORT demonstrate strong performance in object tracking, yet few studies have applied them specifically in child density estimation for urban planning. Additionally, while site selection models emphasize business optimization, they rarely incorporate demographic-specific crowd insights—such as child presence—which could significantly enhance planning for child-focused infrastructure [29–31].

III. IoT Architecture

In this section, we present our proposal: the IoT architecture, which consists of four layers, as shown in Fig. 1.

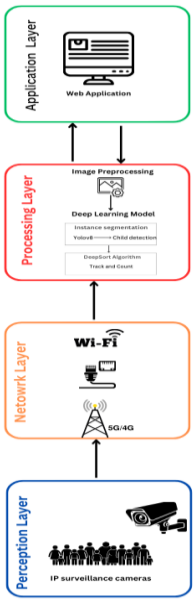


Fig. 1. IoT architecture.

First, the perception layer serves as the base-sensing layer, consisting of sensors that capture real-time environmental data and transmit it to the network layer via various communication protocols. In our work, we exploit the fact that surveillance cameras are widely deployed in public spaces, utilizing them as sensing devices to obtain video frames.

Second, the network layer functions as a communication bridge, connecting layers through internet protocols and network technologies like Wi-Fi, cellular networks, and Ethernet. Wi-Fi offers secure, reliable, and high-speed wireless connectivity, while cellular networks provide wide-area connectivity, supporting high data rates for IoT devices. Ethernet is a wired network that suits real-time applications due to its security and reliability, serving as a backbone network for interconnecting IoT gateways [32]. These features are crucial for sensors in IoT systems. In this work, data are collected from the sensors/cameras and transmitted to the processing layers via the network layer.

Third, the data processing layer processes the collected data from the perception layer through the network layer. This layer is where the intelligence resides, intertwining computer vision and deep learning technologies. Initially, our model preprocesses the frames extracted from the cameras and transmits them to our proposed model. The model then performs analysis and estimation by detecting children, tracking, and counting them accurately. The results generated from the model are then passed to the application layer for practical use.

Finally, the application layer presents the results to the end user through an interactive user interface. In our envisioned application, the results are displayed as a heatmap that ranks different sites based on varying child densities using easy-to-interpret color-coded visualizations.

IV. Methodology

In this section, we outline our methodology to demonstrate how the proposed system was designed, implemented, and evaluated. We started with deploying AI and computer vision tasks, including classification, object detection, and instance segmentation to achieve our objectives. We elaborate on our proposed model as well as other deep learning models in terms of model setup, building, and evaluation. We examine four You Only Look Once (YOLOv8) models, and the highest-performing model is compared against other proposed solutions [1, 39]. We then discuss the tracking and counting mechanism we adopt for estimating child density in crowds.

A. Deep Learning Model Selection

In this part, we start by showcasing the three main elements of any deep learning model for detection: setup, building, and evaluation.

1) Setup: Initially, we searched for a suitable dataset by exploring various open-source platforms. We found a dataset on Kaggle that contained 800 pictures [33]. However, it was not ideal as it did not meet our model's requirements and yielded poor results. Consequently, we created a new dataset by collecting data from public places and live streaming videos from a YouTube channel named "The Real Samui Webcam" [34]. Finally, we cleaned the data and extracted the frames that met our model's requirements.

After collecting the data, we began preprocessing it for the model. First, we classified the dataset into "child" and "non-child," as our focus lies between these two classes. Subsequently, we utilized Roboflow to apply data augmentation techniques, such as resizing images to 640 × 640 pixels and grayscale conversion for 15% of the images to help the model generalize across different lighting conditions. To further enhance the model's adaptability, we increased the saturation of the images by up to 30%, allowing it to learn effectively in various lighting scenarios throughout the day. Finally, to ensure data robustness, we split the dataset into 80% for training, 10% for validation, and 10% for testing.

2) Model building: Our strategy was to deploy different types of models to ensure optimal results. We began with a traditional CNN and applied the classification task among the different layers. However, the results were unsatisfactory. We then shifted our focus to transfer learning, a machine learning approach that uses previously acquired knowledge to solve a related problem. We searched for a suitable pretrained model and chose YOLO, a real-time object detection system [35]. YOLO is a widely used object detection and image segmentation model known for its speed and accuracy [36]. However, it has many versions, and we opted for YOLOv8, which introduced new features and improvements to enhance performance, flexibility, and efficiency.

YOLOv8 allows for multiple computer vision tasks, such as object detection and instance segmentation. We began with

object detection and annotated the images in the dataset with bounding boxes. Then we imported our dataset into the model. Finally, we tested our model to assess the outcomes and discovered that they needed improvement. We attempted to enhance the results by increasing the number of images in the dataset, but the results remained unsatisfactory.

Delving deeper into YOLOv8, we decided to experiment with other computer vision tasks. The first was semantic segmentation, which assigns a class label to pixels using a deep learning algorithm. It is one of three subcategories in the overall process of image segmentation that helps computers understand visual information. However, after trying it, we found that the task's process could not accomplish our goals because it aims to accurately classify each pixel in an image according to the relevant object category, meaning it cannot distinguish between distinct instances of a single object class [37]. Therefore, we decided to try the second task, instance segmentation. This method identifies and separates individual objects within an image, including detecting the boundaries of each object, assigning a unique label to each object, and distinguishing between distinct instances of a single object class. Due to their different performance, we precisely tuned two types of YOLOv8-seg—YOLOv8s-seg and YOLOv8x-seg—to choose the best performer among them. YOLOv8s-seg is smaller and used for faster inference, containing 261 layers. In contrast, YOLOv8x-seg is a larger, more complex model designed to achieve higher segmentation accuracy, with 401 layers, making it slower in inference speed and more costly. By incorporating it into our process, we achieved the best results.

For the training process, we unified the hyperparameters as shown in Table I. The epochs parameter is set to 100, defining the maximum number of iterations during training. The learning rate (lr0) is set to 0.00001, influencing the rate at which the model updates its parameters based on the training data. The batch size is set to 16, determining the number of training samples processed before the model's internal parameters are updated. Using these configurations along with other hyperparameters, we initiated training, ensuring a consistent and effective evaluation process.

TABLE I. HYPERPARAMETERS SETTINGS

Number of Iterations	100 Epochs
learning rate	0.00001
batch size	16
image size	640 × 640 pixels

3) Evaluation: We built multiple models to achieve optimal performance. Table II showcases the diverse results of five model variations in terms of four performance metrics. The mAP@50 is a metric used to evaluate the performance of object detection and instance segmentation models. It measures how well a model can identify and localize distinct objects within an image. The mAP@50 is calculated as the mean of the average precision values across all object classes in the dataset, providing a comprehensive measure of a model's ability to detect and classify distinct object types accurately. Precision, measures the accuracy of the model's positive predictions; therefore, the higher the value, the better the result, indicating that the model accurately detects the object within its correct class and avoids misclassification. Lastly, the recall metric refers to the ratio of correct values predicted relative to all correct values, clarifying the model's completeness in identifying the most relevant instances. As for the models, we first built a CNN with a classification architecture in which the nm3467output is a set of probabilities for each class (child and non-child), with the highest probability indicating the predicted class within a separate frame.

TABLE II. MODELS COMPARISON

Model	Accuracy	mAP@50	Precision	Recall
CNN Classification	0.24	-	0.537	0.598
Object Detection by (YOLOv8s-detect)	-	0.911	0.821	0.873
Object Detection by (YOLOv8x-detect)	-	0.903	0.837	0.907
Instance Segmentation by (YOLOv8s-seg)	-	0.985	0.967	0.954
Instance Segmentation by (YOLOv8x-seg)	-	0.99	0.972	0.988

However, it does not detect two classes in one frame, leading to inaccurate results. Moreover, as shown in Table II, it has poor accuracy and precision of 0.24 and 0.537, respectively. The recall of 0.598 indicates that the model cannot accurately detect and distinguish the classes. In contrast, the object detection model is commonly used for tasks involving detecting multiple classes of objects in the same frame. Thus, we decided to build a model based on object detection. However, detecting objects with a bounding box showed better results than CNN. We tried multiple pretrained models for YOLOv8 object detection that Ultralytics offers [38]. We started with a small pretrained model (YOLOv8s-detect), which is lightweight and makes the trained model fast and efficient due to its limited computational resources. The result was still not as expected because the boxes could not precisely detect the full individual body shape in all positions. The mAP@50 was 0.911, precision increased to 0.821, and recall reached 0.873. Subsequently, we tried the extra-large model (YOLOv8x-detect), which is slower than the small model but more accurate and complex, as it has more layers and achieves higher performance. The result from the YOLOv8x-detect for the mAP@50 was 0.903, and precision and recall increased to 0.837 and 0.907, respectively, compared to the previous model.

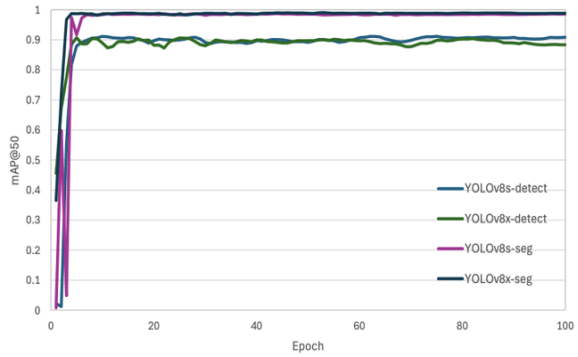
To explore deeper knowledge and acquire more accurate results, we tried another computer vision task that assigns a class label to pixels using a deep learning algorithm, which is instance segmentation. Instance segmentation provides fine-grained localization, enabling precise differentiation between individuals in various positions. This method provides detailed and accurate information, making it easier to distinguish between different objects within an image.

We also tried multiple pretrained models for YOLOv8 instance segmentation. First, we started with a small pretrained model (YOLOv8s-seg), which is lightweight and makes the trained model fast and efficient. Fortunately, the metrics achieved drastic improvements compared to the previous models, with the mAP@50 increasing to 0.985, and precision and recall reaching 0.967 and 0.954, respectively. Then we moved to the extra-large model (YOLOv8x-seg), which is more accurate and complex. The results of the YOLOv8x-seg model yielded the best results, with a mAP@50 of 0.99 and a precision of 0.972. The recall showed the most substantial improvement, reaching 0.988, confirming that the extra-large model outperformed all prior variations.

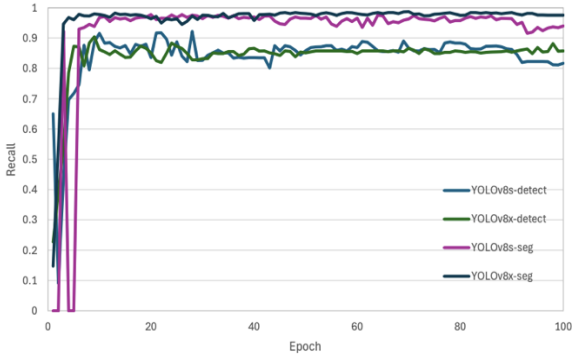
Fig. 2 depicts the performance evaluation of the four YOLOv8 model variations over a hundred epochs based on three key metrics—precision, recall, and mAP@50. Generally, the YOLOv8x models demonstrated more consistent and reliable performance compared to the YOLOv8s models across both the object detection and instance segmentation tasks. The advantage of YOLOv8x is its architectural design. The YOLOv8x model has more layers and a larger set of parameters compared to the YOLOv8s model. This increased complexity enables the YOLOv8x model to learn more effectively, allowing for a better understanding of the underlying data. However, it is important to consider that these complex computations consume more resources. For that reason, we initially used the YOLOv8s model, which proved to be a solid alternative.

YOLOv8s-detect and YOLOv8x-detect models achieve precision, recall, and mAP@50 around 0.7 to 0.8, 0.6 to 0.8, and 0.8 to 0.9, respectively. Among the two, YOLOv8x-detect's results were much closer but smoother. On the other hand, YOLOv8s-seg and YOLOv8x-seg achieved precision, recall, and mAP@50 up to 0.9 in all metrics. Moreover, the YOLOv8x-seg exhibited a consistent level without significant fluctuations. Overall, the segmentation model YOLOv8x-seg surpasses the detection models by achieving higher and more stable levels across all metrics, making it more reliable for our tasks, which require precise object identification to distinguish between child and non-child.

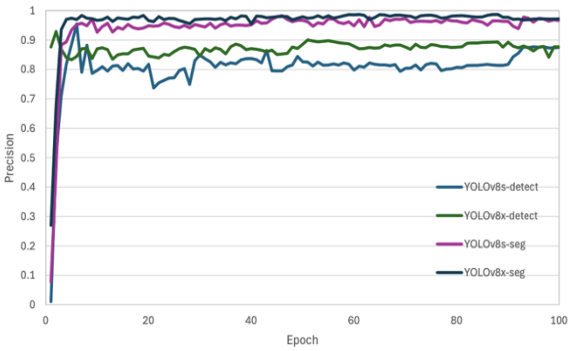
Fig. 3 depicts the relationship between precision and confidence during validation while comparing YOLOv8x-detect, YOLOv8s-detect, YOLOv8s-seg, and YOLOv8x-seg models. Initially, YOLOv8x-detect exhibits a precision of approximately 0.38, which progressively ascends, stabilizing around 0.8, and later increases to approximately 0.92. YOLOv8s-detect starts with a precision range of 0.2 to 0.3. This model then demonstrates a consistent rise to approximately 0.8 to 0.89. There is an apparent rise in precision to the range of 0.8 to 0.9, followed by a minor drop and a subsequent clear increase to around 0.95. From these observations, we infer that YOLOv8s-detect has higher confidence values but slightly lower precision compared to YOLOv8x-detect. In contrast, YOLOv8s-seg surpasses both detection models in precision and confidence, starting at approximately 0.56 and steadily increasing to nearly 0.88, followed by a gradual increase to nearly 0.98. Finally, YOLOv8x-seg outperformed all other models, particularly in terms of precision. It started at approximately 0.8, followed by a gradual increase to 0.9, stabilizing between 0.85 and 0.95, and then ultimately reaching a precision of nearly one.



(a)



(b)



(c)

Fig. 2. YOLOv8 metrics evaluation: (a) mAP@50, (b) Precision, (c) Recall.

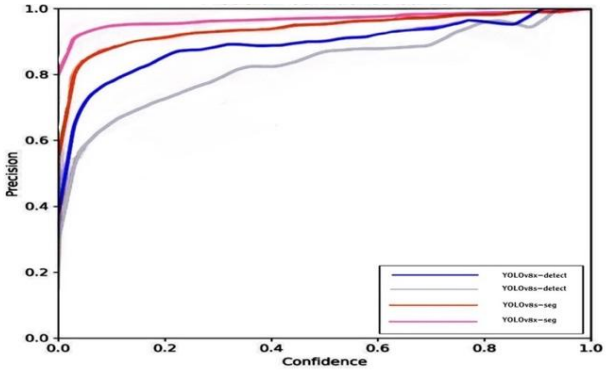


Fig. 3. Precision and confidence in validation.

4) Performance comparison with other YOLO-based models: We compare our highest achieving model (YOLOv8x-seg) as explained previously to other solutions proposed by Tahir et al. [1] and Lin et al. [39]. Fig. 4 compares the performance of these three YOLO-based models: YOLOv4-tiny [39], YOLOv5s [1], and YOLOv8x-seg.

Subfigure (a) illustrates the overall comparison across the mAP@50, Precision, and Recall metrics, while (b–d) provide a detailed view of each metric. Among the models, YOLOv8x-seg achieves the best performance, with a mAP@50 of 0.99, Precision of 0.972, and Recall of 0.988. These exceptional results are attributed to its advanced architectural design and segmentation optimization, which enhance both accuracy and detection capabilities.

The YOLOv5s [1] model also demonstrates high performance, achieving a mAP@50 of 0.905, Precision of 0.879, and Recall of 0.877. While it performs well overall, it remains slightly behind YOLOv8x-seg. Additionally, its effectiveness decreases under suboptimal camera angles, limiting its ability to detect objects accurately in certain scenarios. In contrast, YOLOv4-tiny [39] delivers the lowest scores, with a mAP@50 of 0.652, Precision of 0.83, and Recall of 0.59. These results reflect its performance limitations, especially in challenging tasks such as detecting small objects like babies in strollers, where it struggles to generate precise bounding boxes.

Overall, YOLOv8x-seg stands out as the most advanced and accurate model, making it the ideal solution for applications where performance and precision are critical.

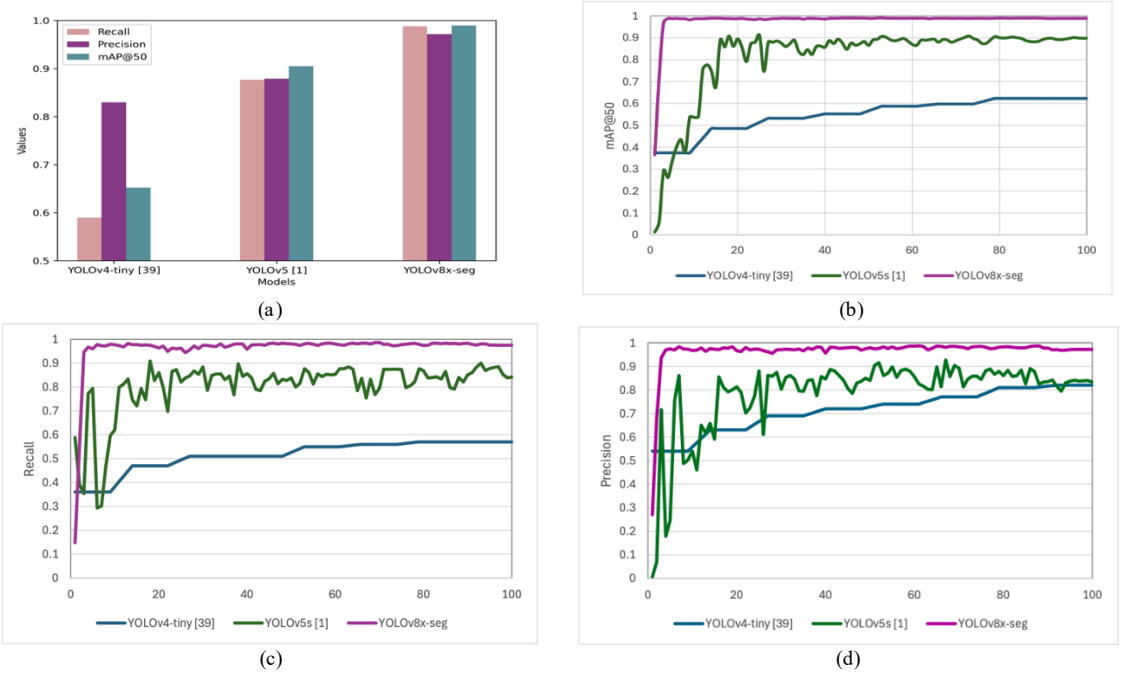


Fig. 4. Comparison of three YOLO-based models: (a) All metrics, (b) mAP@50, (c) Recall, (d) Precision.

B. Tracking and Counting Algorithm

The process of tracking objects across multiple frames is known as object tracking. In object tracking, a unique ID is assigned to each detected object and tracked across multiple frames. Object tracking has gained popularity with the evolution of computer vision and deep learning techniques [40]. Once we successfully applied our model and detected children in an area, we used object- tracking algorithms to follow them. We employed the highly effective DeepSORT algorithm [24], which uses the Kalman filter to track the position and motion of objects over time. The DeepSORT algorithm keeps track of each detected child as they move through the scene. Additionally, several parameters play a critical role in optimizing the tracking and counting process. We set these parameters to achieve optimal results. One such parameter is "n_int" which specifies the minimum number of consecutive frames a child must appear in to be considered for tracking and assigned a unique ID. We set this value to three, meaning that a child must be present in three consecutive frames to ensure its correct presence in the scene before being tracked and assigned an ID.

Another important parameter affecting tracking and counting is "max_age". It defines the maximum number of frames a child can miss before its ID is considered lost and subsequently removed. We set max_age to seventy frames, as it provides a good balance between maintaining persistent tracks and quickly removing children who have left the scene. Finally, after the tracking process is initiated, we count number of the unique IDs of the detected children. If the total count in a given area is high, it is considered a child-crowded area.

V. Results

The implementation had two phases. The first phase involved instance segmentation, which utilized the YOLOv8x-seg model.

We evaluated the model's performance on the training and validation datasets. As a result, the model successfully

identified and classified various objects in the images with high accuracy. Fig. 5 illustrates the experimental results achieved by the model using three metrics: precision, recall, and mAP@50. We also evaluated the model's performance on a new set of images outside the training set. Remarkably, our model demonstrated high accuracy, indicating that it was well-trained and capable of recognizing and differentiating between children and non-children in images. Fig. 6 represents the confusion matrix, a fundamental tool for evaluating performance. It provides a detailed representation of the model's predictions compared to the actual data. We used the confusion matrix to calculate crucial performance metrics such as precision and recall. We concluded that utilizing the YOLOv8x-seg model was the right decision, as it is designed for the instance segmentation task, enabling accurate object detection and unique label assignment, which helps with object counting.

The second phase involved tracking and counting the detected objects. Based on the results obtained from our model, we incorporated the DeepSORT tracking algorithm into our implementation. It enables us to track the children after the model detects them. The results appear as a bounding box around each child, with the track ID and confidence number. Each time a child is detected, the count increases by one. The count, along with the camera's geographical location, is then recorded. These locations near each other represent an area. If the total count in an area is high, it is considered child-crowded and thus optimal for planning child-centric sites such as schools, centers, parks, etc. This approach provides valuable insights into the density and distribution of children within a monitored area.

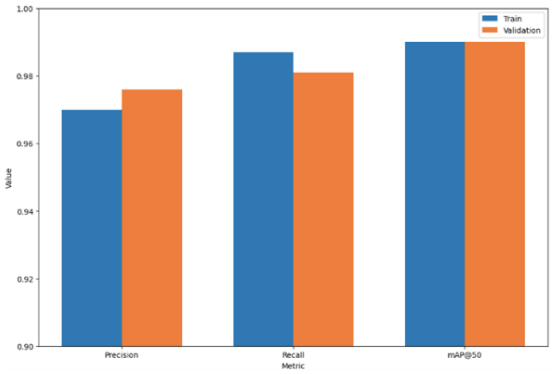


Fig. 5. Metrics comparison of training and validation.

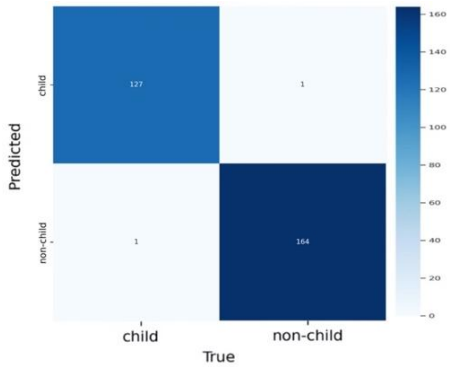


Fig. 6. Confusion matrix.

VI. Conclusion

In a time shaped by technological advancements aimed at improving the quality of life in smart cities; modern tools and techniques have become prevalent across various domains. This study targets improving site selection, assisting entrepreneurs in identifying optimal locations for their businesses while enabling governments to promote economic growth and safety. We propose integrating IoT, YOLOv8, and computer vision techniques to detect, track, and count children in crowds. This paper not only demonstrates improved site selection for entrepreneurs but also highlights broader applications, such as enhancing public safety, improving demographic data collection, and supporting smart urban planning. Our findings underscore the significant impact of our study on urban planning initiatives targeting children. Our contributions illustrate the transformative potential of IoT and AI in addressing societal challenges and fostering economic growth by providing actionable insights for child-focused site optimization. Consequently, we propose an IoT architecture that collects frames from surveillance cameras and processes them using deep learning and computer vision techniques.

Our primary contribution revolves around the IoT processing layer, implemented in two phases: the YOLOv8x-seg model and the DeepSORT tracking algorithm. Our implementation accurately identifies and classifies children in images with a precision rate of 98%. Utilizing emergent techniques such as IoT and deep learning can yield valuable insights that assist businesses and governments in making informed decisions. However, it is crucial to consider the ethical implications, particularly when dealing with sensitive data related to child tracking and monitoring. Although we propose a model for predicting optimal areas for child-centric sites, its deployment must operate within a regulated governance framework that ensures data privacy. Ultimately, the AI-driven system for defining optimal areas for governments and businesses represents a powerful tool that aligns technology utilization with community needs.

References

[1] Tahir, A, Khalid, S.K.A, Fadzil, L.M. "Child Detection Model Using YOLOv5". J. Soft. Compute. Data Min. 2023, 4, 72–81.

[2] Basaran, C, Yoon, H.J, Ra, H.K, Son, S.H, Park, T, Ko, J. "Classifying children with 3D depth cameras for enabling children's safety applications". In proc. of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Seattle, WA, USA, 13–17 September 2014; pp. 343–347.

[3] Balci, B, Alkan, B, Elihos, A, Artan, Y. "Front seat child occupancy detection using road surveillance camera images". In proc. of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 1927–1931.

[4] Kaiser, M.S, Lwin, K.T, Mahmud, M, et al. "Advances in Crowd Analysis for Urban Applications Through Urban Event Detection". IEEE Trans. Intell. Transp. Syst. 2018, 19, 2518–2531. https://doi.org/10.1109/TITS.2017.2771746.

[5] Weda, H, Barbieri, M. "Automatic children detection in digital images". In proc. of the 2007 IEEE International Conference on Multimedia and Expo, Beijing, China, 2–5 July 2007; pp. 1687–1690.

[6] Xu, M, Wang, T, Wu, Z, Zhou, J, Li, J, Wu, H. "Demand Driven Store Site Selection Via Multiple Spatial-Temporal Data". In proc. of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Burlingame, CA, USA, 31 October–3 November 2016; pp. 1–10. https://doi.org/10.1145/2996913.2996996.

[7] Mouha, R.A. “Internet of Things (IoT)”. J. Data Anal. Inf. Process. 2021, 9, 77–101. https://doi.org/10.4236/jdaip.2021.92006.

[8] Shambour, M.K, Gutub, A. “Progress of IoT Research Technologies and Applications Serving Hajj and Umrah”. Arab. J. Sci. Eng. 2022, 47, 1253–1273.

[9] Krishnamurthi, R, Kumar, A, Gopinathan, D, Nayyar, A, Qureshi, B. “An Overview of IoT Sensor Data Processing, Fusion, and Analysis Techniques”. Sensors 2020, 20, 6076. https://doi.org/10.3390/s20216076

[10] Alam, T. “The Relation of Artificial Intelligence With Internet Of Things: A survey”. ResearchGate 2020, 1, 30–34.

[11] McCarthy, J. “What Is Artificial Intelligence?” Stanford University: Stanford, CA, USA, 2007. Available online: http://jmc.stanford.edu/articles/whatisai.html (accessed on 12 June 2024).

[12] Varghese, E.B, Thampi, S.M. “Application of Cognitive Computing for Smart Crowd Management”. IT Prof. 2020, 22, 43–50.

[13] IBM. “What Is Computer Vision?” Available online: https://www.ibm.com/topics/computer-vision (accessed on 3 May 2024).

[14] Chan, A.B, Vasconcelos, N. “Crowd Analysis Using Computer Vision Techniques”. IEEE Signal Process. Mag. 2010, 27, 66–77. https://doi.org/10.1109/MSP.2010.937394.

[15] Solmaz, G, Wu, F, Cirillo, F, et al. “Toward Understanding Crowd Mobility in Smart Cities through the Internet of Things”. IEEE Signal Process. Mag. 2019, 57, 40–46. https://doi.org/10.1109/MCOM.2019.1800611.

[16] Altwayan, L, AlMuhayfith, M, Alshahrani, et al. “An Intelligent IoT Approach for Analyzing and Managing Crowds”. IEEE Access 2021, 9, 104874–104886. https://doi.org/10.1109/ACCESS.2021.3099531

[17] Li, L. “A Crowd Density Detection Algorithm for Tourist Attractions Based on Monitoring Video Dynamic Information Analysis”. Complex. J. 2020, 2020, 14. https://doi.org/10.1155/2020/6635446.

[18] Wong, V.W.H, Law, K.H. “Fusion of CCTV Video and Spatial Information for Automated Crowd Congestion Monitoring in Public Urban Spaces”. Algorithms 2023, 16, 154. https://doi.org/10.3390/a16030154.

[19] Zhou, Q, Gu, J, Lu, X, Zhuang, F, Zhao, Y, Wang, Q, Zhang, X. “Modeling Heterogeneous Relations across Multiple Modes for Potential Crowd Flow Prediction”. AAAI Conf. Artif. Intell. 2021, 35, 4723–4731. https://doi.org/10.1609/aaai.v35i5.16603.

[20] Chua, S.N.D, Lim, S.F, Lai, S.N, Chang, T.K. “Development of a Child Detection System with Artificial Intelligence Using Object Detection Method”. J. Electr. Eng. Technol. 2019, 13, 2523–2529.

[21] Ince, O.F, Ince İ, F, Park, J.S. “Using biometric features on long-distance videos for accurate pedestrian age classification”. Bilişim Teknol. Derg. 2017, 10, 123–128. https://doi.org/10.17671/gazibtd.309264.

[22] Ince, O.F, Ince, I.F, Park, J.S, Song, J.K, Yoon, B.W. “Child and adult classification using biometric features based on video analytics”. ICIC Express Lett. Part B Appl. 2017, 8, 819–825.

[23] Agarwal, M, Jain, S. “Image Classification for Underage Detection in Restricted Public Zones”. In proc. of the 8th International Advance Computing Conference (IACC), Greater Noida, India, 14–15 December 2018; pp. 355–359.

[24] Bewley, S, Ge, Z, Ott, L, Ramos, F, Upcroft, B. “Simple Online and Realtime Tracking with a Deep Association Metric”. In proc. of the IEEE International Conference on Image Processing, Beijing, China, 17–20 September 2017; pp. 3464–3468. https://doi.org/10.1109/ICIP.2016.7533003.

[25] Wu, J, Wang, X, Huang, L, Wang, Z, Wan, D, Li, P. “Parameterized Site Selection Approach of Park Entrance Based on Crowd Simulation and Design Requirement”. Appl. Sci. J. 2023, 13, 6280, . https://doi.org/10.3390/app13106280.

[26] Alossta, O.E, Badi, I. “Resolving a Location Selection Problem By Means of an Integrated AHP-RAFSI Approach”. ResearchGate 2021, 2, 135–142. https://doi.org/10.31181/rme200102135a.

[27] Alhothali, B.A, Faisal, K, Alshammari, et al. “Location-Allocation Model to Improve the Distribution of COVID-19 Vaccine Centers in Jeddah City, Saudi Arabia”. Int. J. Environ. Res. Public Health 2022, 19, 8755. https://doi.org/10.3390/ijerph19148755.

[28] Akbari, V, Rajabi, M.A, Shams, R, Chavoshi, S.H. “Landfill Site Selection by Combining GIS and Fuzzy Multi Criteria Decision Analysis, Case Study: Bandar Abbas, Iran”. World Appl. Sci. J. 2008, 3, 39-47.

[29] Snieska, V, Zykiene, I, Burksaitiene, D. “Evaluation of Location’s Attractiveness for Business Growth in Smart Development”. Econ. Res.-Ekon. Istraživanja 2019, 32, 925–946. https://doi.org/10.1080/1331677X.2019.1590217.

[30] Akpan, S.J, Uford, I. “Retail Outlet Location and Business Performance in Uyo Metropolis”. Int. J. Adv. Manag. Econ. 2023, 12, 9–18.

[31] Yakovlev, S, Kiseleva, O, Chumachenko, D, Podzeha, D. “Maximum Service Coverage in Business Site Selection Using Computer Geometry Software”. Electronics 2023, 12, 2329. https://doi.org/10.3390/electronics12102329.

[32] Cheruvu, S, Kumar, A, Smith, N, Wheeler, D.M. “Demystifying Internet of Things Security: Successful IoT Device/Edge and Platform Security Deployment”, 1st ed, Apress: Berkeley, CA, USA, 2020.

[33] Farchione, D. Children vs Adults Classification; Kaggle: San Francisco, CA, USA, 2023.

[34] YouTube Channel. The Real Samui Webcam. Available online: https://www.youtube.com/@TheRealSamuiWebcam (accessed on 13 March 2024).

[35] Redmon, J, Divvala, S, Girshick, R, Farhadi, A. “You Only Look Once: Unified, Real-Time Object Detection”. arXiv, 2015, arXiv.1506.02640. https://doi.org/10.48550/arXiv.1506.02640.

[36] Lee, L.J, Desa, H, Azizan, M.A, Hussain, A.T, Tanveer, M.H. “Object Detection and Instance Segmentation with YOLOV8: Progress and Limitations”. Proc. Int. Conf. Artif. Life Robot. 2024, 29, 724–728. https://doi.org/10.5954/icarob.2024.os23-5.

[37] Saidani, T. “Deep Learning Approach: YOLOv5-based Custom Object Detection”. Eng. Technol. Appl. Sci. Res. 2023, 13, 12158–12163.

[38] Ultralytics. Segment Documentation. Available online: https://docs.ultralytics.com/tasks/segment/ (accessed on 3 March 2024).

[39] Lin, J.-M, Lin, W.-L, Fan, C.-P. “Age Group Classifier of Adults and Children with YOLO-based Deep Learning Pre-Processing Scheme for Embedded Platforms”. In proc. of the 12th International Conference on Consumer Electronics (ICCE-Berlin), Berlin, Germany, 2–6 September 2022; pp. 1–5. https://doi.org/10.1109/ICCE-BERLIN56473.2022.9937129.

[40] Sharma, N, Baral, S, Paing, M.P, Chawuthai, R. “Parking Time Violation Tracking Using YOLOv8 and Tracking Algorithms”. Sensors 2023, 23, 5843. https://doi.org/10.3390/s23135843.

D.M.A.I.H.: Deepfake-Inspired Few-Shot Learning Approach with Stable Diffusion for Digital Mourning

Btissam Acim, Hamid Ouhnni, Nassim Kharmoum, Soumia Ziti
Computer Science Department-Intelligent Processing and Security of Systems (IPSS) Laboratory-Faculty of Sciences, Mohammed V University in Rabat, Morocco

***Abstract*—Digital mourning (deuil numérique) is the use of digital and AI-based technologies to preserve, recontextualize, and extend the memory of deceased loved ones through personalized and meaningful virtual representations. The digital mourning process requires innovative technologies capable of preserving the memory of deceased loved ones in meaningful and humanized ways. This paper proposes a novel generative approach, D.M.A.I.H. (Digital Mourning with Artificial Intelligence for Healing), for digital grief, with a focus on moral support and the mental health of bereaved relatives, using Stable Diffusion with a few-shot learning adaptation mechanism. The system takes as input a small set of personal references (e.g. a portrait, contextual images such as the person's home, and a short descriptive script) and outputs high-quality, photorealistic images of the deceased in different meaningful contexts, process closely related to deepfake generation but redirected here toward therapeutic and commemorative purposes. Unlike traditional generative models requiring large datasets, Few-shot personalization is leveraged to adapt Stable Diffusion to each individual with minimal data, enabling the generation of personalized digital albums. Experimental results show that the model consistently preserves identity in the images it produces, and contextual control ensures emotional resonance. In particular, identity similarity scores for the generated images ranged from 0.88 to 0.93, with an average score of 0.91, testifying to strong identity preservation across all outputs. The innovation of this study is a foundation for AI-based memorialization, balancing technological innovation with concerns over privacy, authenticity and cultural sensitivity, and psychological comfort.**

***Keywords*—Stable diffusion; few-shot learning; deepfake; Artificial Intelligence (AI); generative AI; digital mourning**

I. Introduction

Generative Artificial Intelligence (AI) is developing at a fast rate and transforming various industries such as art, education, healthcare [1], [2] and entertainment. Here, one of the newly emerging and yet to be discovered uses is digital mourning, where AI can assist in keeping the memory of deceased loved ones alive and providing psychological and ethical assistance to bereaved families [3], [4]. Traditional modes of commemoration, photographs, videos, or symbolic objects remain crucial but static and limited in their ability to capture the emotional richness and contextual diversity of an individual's life. Generative AI, and in particular the proposed approach, D.M.A.I.H. (Digital Mourning with Artificial Intelligence for Healing), enables a dynamic and customized recontextualization of memory, generating visual forms that can enrich the affective bond and accompany mourning [5].

This area, however, presents several difficulties. Ensuring identity consistency in the generated images is the primary challenge, considering that it is essential to accurately preserve the individual's distinctive characteristics. Limited data availability is also a primary constraint, since there are seldom several personal photos. Moreover, there are psychological and ethical issues to be addressed with sensitivity: while this technology has the potential to boost mental health and well-being through moral and empathic remembering, as substantiated by empirical studies on chatbots such as Woebot [6] and longitudinal human–computer interaction for affective care [7], it must also safeguard dignity and authenticity in commemoration [8].

The proposed approach introduces D.M.A.I.H., an innovative method that leverages enhanced Stable Diffusion with few-shot learning to generate personalized memorial albums from a small number of reference images, contextual backgrounds, and descriptive scripts. Unlike conventional generative models that require large datasets, D.M.A.I.H. adapts efficiently to each individual, ensuring both identity preservation and faithful contextualization [9].

As opposed to existing digital commemoration sites such as Facebook Memorialized Accounts [10], Instagram memorial accounts [11], or websites such as Forever Missed, which allow for mostly static or symbolic remembering, our approach offers a generative pipeline that can generate personalized, contextualized, and photorealistic memorial albums. This more appropriately situates the contribution in the field and highlights its novelty without overstating it.

First, D.M.A.I.H. is introduced as a generative approach inspired by recent advances in deepfake synthesis, but oriented toward ethically grounded memorialization, specifically designed for digital mourning and capable of producing personalized albums with high emotional value. Second, it is demonstrated that the integration of few-shot learning into Stable Diffusion effectively overcomes the limitation of scarce data while maintaining identity consistency. Finally, the D.M.A.I.H. situated within a broader perspective of psychological and moral support, emphasizing the potential of this technology to assist the mental health of grieving families while opening an ethical debate on its responsible use [12].

In brief, the D.M.A.I.H. approach was adopted since it is an approach that addresses well the core issues of digital bereavement, i.e., sparsity of data, identity integrity, and emotional relevance. The existing approaches, though promising, are afflicted by either excessive computational cost,

low identity integrity, or limited contextual adaptability and thus are less suitable for the sensitive nature of loss. On the other hand, D.M.A.I.H. integrates Stable Diffusion and few-shot learning to bypass these limitations to develop a robust generative framework for personalized memorial albums.

The remaining part of this paper is organized as follows: Section II gives an overview of related work, Section III presents the proposed model, Section IV provides experimental results, Section V discusses the findings in the context of previous studies, and Section VI concludes with future directions and ethical implications.

II. Related Work

Generative AI has been marked by an ongoing enhancement of Generative Adversarial Networks (GANs) [13] and diffusion models. *Deepfake* techniques, originally popularized through GAN-based face synthesis, illustrate both the power and the risks of these generative systems. StyleGAN, for example, has performed incredibly well in face generation [14], while Stable Diffusion, a diffusion model, has transformed text-to-image synthesis through its controllability and replication of visuals [15]. In order to fulfill personalization needs with limited data, various approaches have emerged, such as DreamBooth [16], LoRA (Low-Rank Adaptation) [17], and DreamArtist [18]. These approaches facilitate identity preservation with as few as 5 to 15 images, and thus are suitable in situations where the data is not abundant. However, their applications have largely been limited to designing avatars, artwork, or entertainment, and have been investigated extremely sparingly in delicate topics such as loss.

Digital technology has revolutionized commemoration practices, to the extent of producing what has been referred to as digital mourning. Social media such as Facebook Memorialized Accounts [19], or Instagram Memorial Accounts [20], allow surviving relatives to keep deceased individuals' pages and post memories day and night. Memory-keeping websites such as Forever Missed and Legacy.com also provide virtual space for memory preservation.

Alongside these static memorials, AI-driven initiatives have emerged. HereAfter, AI records a person's voice and stories during their lifetime, enabling interactive experiences after death. StoryFile offers similar interactive video testimonies. Deep Nostalgia (MyHeritage) animates old photographs using AI, while projects such as Eternime envisioned the creation of persistent digital avatars. Research prototypes and patents, such as Microsoft's so-called "Deadbots", have also suggested the use of digital traces to simulate posthumous conversations. While these solutions illustrate the growing interest in digital commemoration, they remain limited to static memorials, conversational agents, or simple animations, without providing truly personalized visual albums.

In addition to commemoration, AI is more and more studied for its therapeutic applications in mental health treatment. Therapeutic robots such as Replika are used for emotional care, and virtual avatars have been explored for therapy. Existing studies highlight the ability of AI to foster well-being, empathy, and psychological resilience [21], [22]. These approaches often favor verbal or dialogic interaction, while the visual and contextual dimensions of loss support are largely untapped.

This review demonstrates that while generative personalization methods and digital mourning platforms exist separately, no study to date has proposed a unified pipeline for generating memorial albums using Stable Diffusion with few-shot learning. Current platforms are limited to conversation or static commemoration, whereas AI-based personalization methods remain confined to art and entertainment. The proposed approach addresses this gap by introducing D.M.A.I.H., an innovative generative framework capable of producing photorealistic and contextualized albums of deceased individuals, thereby bridging generative AI research with the psychological needs of digital mourning.

III. Proposed Model

The proposed D.M.A.I.H. approach leverages the generative capacity of Stable Diffusion, enhanced with few-shot learning, to create personalized memorial albums. Stable Diffusion was selected as the backbone of the proposed approach because it combines photorealistic image generation with efficient conditioning mechanisms, enabling both identity preservation and contextual control. Compared to GAN-based models, which must operate on vast datasets and are often beset with instability (e.g. mode collapse), diffusion models and Stable Diffusion are more stable, more robust, and more scalable for supervised image creation. While deepfake techniques have traditionally relied on GAN-based architectures, the D.M.A.I.H. approach demonstrates how diffusion models can achieve similar identity-focused synthesis while being redirected toward therapeutic and commemorative purposes.

The whole pipeline of the D.M.A.I.H. approach is illustrated in Fig. 1, where identity extraction, contextual conditioning, and text guidance are integrated into the U-Net backbone of Stable Diffusion.

A. Approach Overview

The proposed D.M.A.I.H. approach is designed to address three key challenges in digital mourning: data scarcity, identity preservation, and emotional relevance**.** The system takes as input a small set of target images of the deceased, contextual scenes representing meaningful environments (e.g. home, farm, or interior spaces), and a descriptive script provided by relatives.

These heterogeneous inputs are processed by specialized encoders and conditioning modules, then fused within the denoising process of Stable Diffusion's U-Net. The result is the generation of multiple high-quality, photorealistic images which collectively form a personalized memorial album that preserves both identity fidelity and contextual meaning.

B. Detailed Description of Model Blocks

1) Target (Identity extraction): The reference images of the deceased are first segmented to isolate the face. A Face Encoder (ArcFace or InstantID) extracts identity tokens that encode the distinctive visual traits of the individual, ensuring faithful identity preservation [23].

2) Scene (Contextual integration): Contextual images are processed with ControlNet with depth, Canny edges, and pose

constraints. This guarantees realistic integration of the target subject into meaningful environments [24].

3) Descriptive script (Semantic guidance): A descriptive script provided by relatives is embedded using CLIP/BERT. These embeddings introduce semantic and emotional guidance [25], [26], reinforcing contextual fidelity and emotional resonance in the generation process.

4) Fusion in U-Net (Stable diffusion): Identity tokens, contextual constraints, and text embeddings are fused during the denoising process of the U-Net backbone within Stable Diffusion, yielding coherent and photorealistic outputs.

5) Output (Memorial album): The system produces multiple personalized images under different contexts. These are assembled into a digital memorial album, preserving both the identity of the deceased and the emotional significance of the scenes.

ArcFace was selected due to its state-of-the-art performance on face recognition benchmarks such as LWF (>99.8%) [27], ensuring robust identity preservation. CLIP [28] has been demonstrated to possess robust text–image alignment on large-scale datasets, enabling coherent semantic composition. ControlNet [29] has demonstrated improved performance on controllable generation tasks (pose, depth, edge), which is suitable for contextual fidelity in digital mourning.

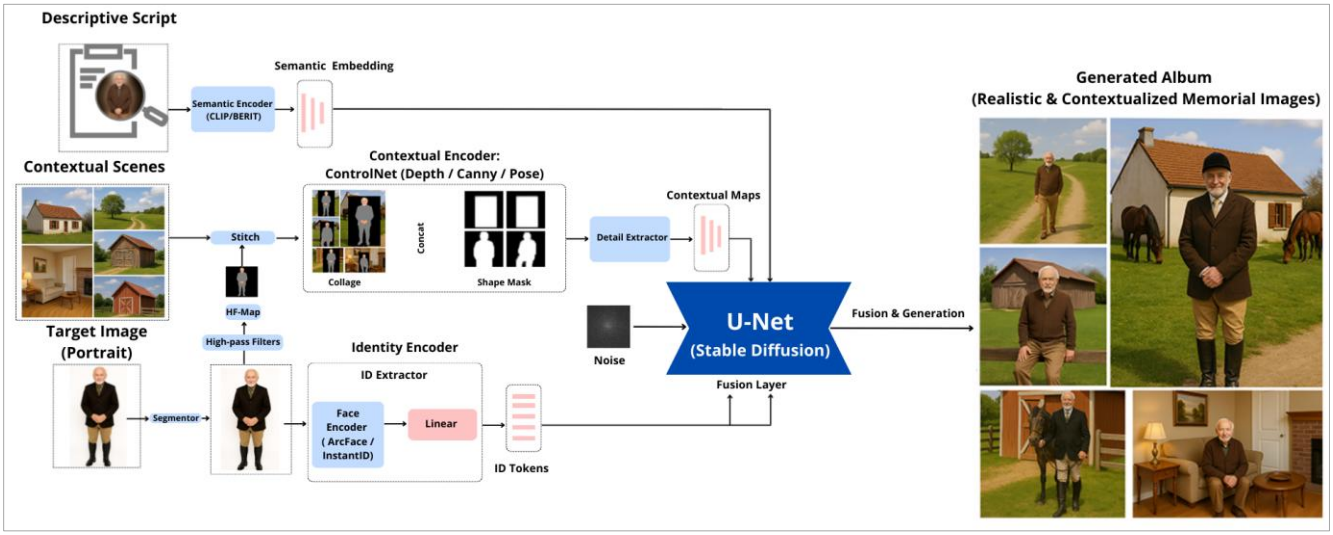


Fig. 1. Architecture of the D.M.A.I.H. approach for personalized digital mourning.

C. Motivation for Choosing Stable Diffusion and U-Net

Several personalization techniques have been explored in the literature:

1) DreamBooth: provides strong identity preservation but requires costly fine-tuning for each new subject.

2) LoRA: reduces training costs but may sacrifice fine-grained identity fidelity.

3) Textual inversion: is lightweight but limited in flexibility and often degrades image quality.

Conversely, Stable Diffusion's U-Net backbone supports [23], the straightforward combination of multiple conditioning sources (identity, context, and text) without full retraining. This balance of fidelity, controllability, and efficiency makes it ideally suited to few-shot personalization in digital bereavement, where personal images are minimal.

It was observed that this is the first such use of Stable Diffusion with few-shot learning [30] in the context of digital mourning, attempting to create photorealistic, contextualized, and emotionally evocative memorial albums.

This is also novel in technique, but it also has a profound psychosocial connotation. By producing coherent and personalized visual representations of the deceased, the system moves beyond image creation to the design of meaningful artifacts that can play a role in the grieving process.

Unlike static images, which are frozen in time, the designed memorial albums create an active and contextualized redesign of memory, allowing families to reopen contact with their loved ones in emotionally significant spaces.

Such recontextualization may have the effect of consolidating the affective bond with the deceased, conferring comfort and continuity upon the process of grieving. It does indeed support the presentation of grief in a visual and narrative form that is consonant with traditional rituals of commemoration. Such personalized albums can thus function as therapeutic agents, facilitating the process of coming to terms and lessening the sense of loss and lack.

From a psychological perspective, this intervention positions generative AI not only as a creative tool but also as a potential aid for mental health and emotional stability. By enabling families to preserve identity, remember the past, and create new symbolic meaning, the proposed D.M.A.I.H. approach can act as a moral and empathic companion bridging technological innovation with fundamental human values of memory, dignity, and resilience.

D. Comparative Analysis of Methods

The points out that while DreamBooth and LoRA give promising results, they are either computationally expensive or less accurate in terms of maintaining identity. The discussed U-Net-based Stable Diffusion pipeline ensures the best trade-off between computational expense, identity fidelity, and context flexibility, which renders it particularly well-suited for digital grief purposes. To further illustrate this comparison, Table I provides a structured overview of the main methods in terms of data requirements, training cost, identity preservation, context control, and suitability for digital mourning.

TABLE I. COMPARATIVE ANALYSIS OF GENERATIVE PERSONALIZATION METHODS FOR DIGITAL MOURNING

Method	*Data Requirement*	*Training Cost*	*Identity Preservation*	*Control Suitability for Digital Mourning*
DreamBooth	5–15 images	High (fine-tuning)	Strong	Limited (costly per subject)
LoRA	5–10 images	Moderate	Moderate	Possible but less precise
Textual Inversion	3–5 images	Low	Weak to Moderate	Not optimal (loss of fidelity)
D.M.A.I.H. approach	5–15 images	Low to Moderate	Strong (0.88–0.93, avg. 0.91)	Highly suitable

As shown in Table I and supported by quantitative validation. (0.88–0.93 identity similarity scores), D.M.A.I.H. achieves greater identity preservation and contextual fidelity than previous approaches.

IV. EXPERIMENTAL RESULTS

To validate the effectiveness of the proposed D.M.A.I.H. approach, an experimental use case representative of the digital mourning process was designed. This case relies on three types of inputs:

- Reference portrait of an elderly woman.
- Contextual scenes representing meaningful environments (family house, farm, interior).
- A descriptive script providing semantic and emotional guidance.

These inputs were integrated into the pipeline, which combines Stable Diffusion with few-shot learning and the U-Net backbone, to generate a personalized memorial album.



Fig. 2. Experimental use case of the D.M.A.I.H. approach.

Fig. 2 illustrates this experimental setup. On the left, the inputs are shown (portrait of the woman, contextual scenes, and descriptive script); in the middle, the Stable Diffusion with few-shot learning process is represented; and on the right, the output is displayed as a five-image memorial album, where the deceased is realistically recontextualized into different meaningful contexts.

The generated images demonstrate the system's ability to preserve identity fidelity while ensuring contextual relevance and emotional resonance, confirming the suitability of the proposed approach for digital mourning applications.

In the experiments, the scalability of the proposed approach was evaluated in generating a large number of personalized images per subject. Specifically, it was possible to produce more than 200 distinct images while maintaining satisfactory identity and contextual fidelity. This result demonstrates the practical scalability of the D.M.A.I.H. approach and its relevance for creating enriched memorial albums. However, this generative capacity remains subject to several technical constraints, including the available GPU memory, the inference time required to produce large batches of images, and the quality of input data, which directly affects the coherence and emotional relevance of the generated outputs. To additionally challenge the

generalizability of D.M.A.I.H., new albums were generated with alternate reference sets, with stable identity preservation and contextual integration. For comparison as a baseline, we applied DreamBooth [31] to the same inputs. While DreamBooth produced visually realistic outputs, it performed worse in terms of contextual fidelity, particularly in grounding descriptive scripts in generated scenes. D.M.A.I.H., by comparison, resulted in higher identity similarity scores (avg. 0.91) and improved contextual coherence.

Our findings are consistent with prior reports of trade-offs between identity fidelity and contextual responsiveness in personalization methods. The results extend these findings by demonstrating that D.M.A.I.H. achieves a more optimal balance, particularly for sensitive applications such as digital mourning.

A. Qualitative Analysis

The output images show that the identity of the subject is preserved across all contexts. The facial features of the individual remain consistent, confirming the effectiveness of the identity extraction module. Correspondingly, the contextual integration is also realistic: the subject is depicted believably in the scenes provided (house, farm, interior), achieving semantic coherence between inputs and outputs. The descriptive script also emphasizes the emotional impact, as it guides the generation to representations that achieve high affective value.

As a result, the album produced is an active visual remembrance rather than static common images. By representing them in several contextualized and emotionally relevant ways, it allows families to symbolically reconnect with deceased loved ones.

B. Quantitative Analysis: Identity Fidelity

In addition to the qualitative analysis, the identity fidelity between the reference image and the generated images of the album was evaluated. For this purpose, A face encoder (ArcFace) was used [24] to extract embeddings and compute the cosine similarity defined as follows:

$$S(I_{ref}, I_{gen}) = \frac{f(I_{ref}) \cdot f(I_{gen})}{\| f(I_{ref}) \| \| f(I_{gen}) \|}$$

where, $f(I)$ denotes the embedding of an image, I_{ref} is the reference image, and I_{gen} is a generated image. A value close to 1 indicates strong identity preservation.

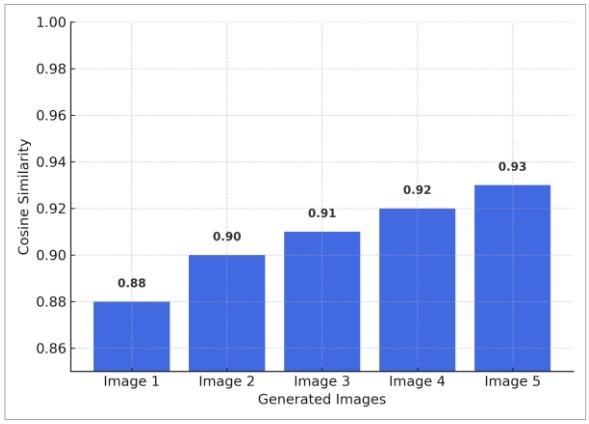


Fig. 3. Cosine similarity scores of generated images (D.M.A.I.H.).

Fig. 3 reports the scores obtained for the five generated images of the album. Results ranging between 0.88 and 0.93, with an average similarity score of 0.91, further demonstrate consistent identity preservation and confirm that the D.M.A.I.H. approach achieves high identity consistency despite the limited number of input images. These validation measures are necessary to demonstrate robustness. By combining quantitative analysis (identity similarity scores) and qualitative analysis (contextual and semantic coherence), the generated outputs are ensured to be not only visually acceptable but also retain the emotional and memorial value required in digital mourning software.

C. Dual Scope of the D.M.A.I.H. Approach

On the technical level, D.M.A.I.H. introduces, for the first time in the field of digital mourning, the combination of Stable Diffusion with few-shot learning. This synergy enables the creation of photorealistic, contextualized, and highly personalized memorial albums from only a very limited number of reference images. By leveraging generative AI with advanced adaptation techniques, the system overcomes the traditional limitations of data scarcity in sensitive contexts such as mourning.

On the psychosocial level, D.M.A.I.H. transcends the boundaries of image generation to become a medium of moral and psychological support. By offering authentic and emotionally resonant representations, it strengthens the symbolic continuity of grief, consolidates the bond between families and the deceased, and complements rather than replaces traditional mourning rituals.

Overall, this paper presents generative AI both as a driver of technological innovation and as an instrument of psychological resilience, emotional well-being, and mental health. It highlights the possibility of an empathetic and ethically responsible use of AI in supporting families, where advanced technology is balanced with fundamental human values such as dignity, memory, and humanity.

V. Discussion

The results obtained highlight the effectiveness and relevance, both from a technical and a psychosocial perspective.

A. Technical Strengths

The D.M.A.I.H. approach, based on Stable Diffusion with few-shot learning, demonstrated its ability to robustly preserve identity fidelity, even with a very limited number of reference images. Similarity scores (0.88–0.93) confirm high identity consistency across generated outputs. Additionally, contextual scene composition using ControlNet ensures realistic placement of the subject in semantically relevant contexts, and the script description ensures semantic and emotional consistency. Compared to existing personalization methods, such as DreamBooth, LoRA, or Textual Inversion, A more favorable trade-off between visual quality is achieved, flexibility, and computational efficiency, without requiring expensive retraining for each new subject, and illustrates how techniques often linked to deepfake generation can be redirected toward ethical applications.

In particular, compared to DreamBooth [32], Costly fine-tuning was required, and weaker contextual fidelity was observed in the baseline implementation; higher identity similarity scores (average 0.91) were achieved. Similarly, while LoRA [31] and Textual Inversion [33] provide lightweight alternatives, they often suffer from degraded identity fidelity or reduced image quality. D.M.A.I.H. therefore offers a stronger balance of identity preservation, contextual adaptability, and computational efficiency, which is especially relevant in sensitive contexts such as bereavement. New innovations such as Perfusion [34] have attempted to reduce fine-tuning costs but keep personalization potential high; however, they fall short regarding identity preservation. In contrast, the model is stronger and more contextual in design.

B. Comparison with Existing Approaches

Current online mourning services, such as Facebook Memorialized Accounts, Forever Missed, or Legacy.com, just offer static memorials. AI-based initiatives such as HereAfter AI, StoryFile, or Deep Nostalgia focus mainly on interactive dialogue or simple photo animation [35]. While innovative, these tools do not provide a comprehensive generative framework capable of producing personalized, contextualized visual albums. The D.M.A.I.H. approach fills this gap by introducing a pipeline that combines identity preservation, contextual control, and emotional resonance into a single system. Moreover, by explicitly stating similarity of identity and correspondence of context, the analysis extends beyond qualitative judgment to provide quantifiable proof of resilience. This not only justifies the comparative analysis by virtue of being against other existing personalization methods, but it positions D.M.A.I.H. as not merely a theoretical contribution but as an empirically proven framework aligned with existing research.

C. Psychosocial Impact

Beyond technical innovation, the main strength the D.M.A.I.H. approach lies in its role as a potential tool for moral and psychological support. The generated albums allow families to symbolically reconnect with their deceased loved ones, transcending the limits of traditional static photographs [36]. This recontextualization of memory as a dynamic process can be an additional therapeutic tool supporting the mourning process, reinforcing emotional stability, and enabling psychological coping in bereaved families.

D. Limitations and Future Perspectives

Despite the positive outcomes of the D.M.A.I.H. approach, it is important to recognize a number of its limitations. Generalizability is limited because the evaluation is still primarily qualitative and based on a single use case. To evaluate the model's robustness, a larger-scale validation process involving user studies in various cultural contexts will be necessary. The integration of multimodal memories (voice, video, and text), the reinforcement of ethical control mechanisms across the pipeline, and the creation of safeguards to reduce risks like emotional dependency are some future research directions. To guarantee greater empirical validation and ethical robustness, the assessment should also be expanded to larger datasets and carried out in conjunction with psychologists and ethicists.

Similar scalability and identity drift issues also have been encountered in recent personalization studies, validating that the proposed extensions support multimodal integration, adversarial robustness, and stronger ethical guardrails are timely and valid directions.

E. Ethical and Legal Considerations

The use of the D.M.A.I.H. approach in the context of bereavement also raises serious ethical and legal concerns. The issue of consent post-mortem remains on everyone's minds, as current legal regulations provide minimal guarantees. The GDPR and upcoming AI Act for Europe impose transparency and traceability obligations on AI systems but only apply partially to posthumous data across jurisdictions. Such risks include the possibility of inducing false memories, emotional manipulation, and psychological dependence among families. In response to these challenges, the D.M.A.I.H. approach is grounded in an ethical responsibility model, viewing the suggested system as a supportive tool supplementary to the natural grieving process rather than a substitute for it. Future efforts must focus on establishing ethical norms and regulatory frameworks to ensure that generative AI is applied in a thoughtful, empathetic, and culturally sensitive manner.

VI. Conclusion

In this paper, the D.M.A.I.H. approach was introduced, a novel pipeline for digital mourning based on Stable Diffusion with few-shot learning, capable of generating personalized, photorealistic, and contextualized memorial albums from a very limited number of reference images. The D.M.A.I.H. approach meets three of the most significant challenges in this area: identity preservation, contextual fidelity, and emotional relevance. Experimental outcomes show that the system proposed herein effectively preserves identity through a change of context, and descriptive scripts ensure semantically and emotionally relevant outputs, illustrating how techniques commonly linked to deepfake generation can be ethically redirected.

Aside from the technical relevance, this study emphasizes the psychosocial significance of the D.M.A.I.H. approach as a tool for moral and psychological attention. Through the dynamic recontextualization of remembrance, the system offers relatives an opportunity to symbolically reunite with their deceased kin, thus complementing everyday acts of memory and encouraging mental health and emotional balance. At the same time, this study identifies important ethical and legal issues, particularly regarding consent, authenticity, and cultural sensitivity. Their resolution will be needed to assure proper use of these technologies.

Unlike existing digital commemoration services that are static or symbolic, D.M.A.I.H. introduces a novel generative pipeline that integrates identity preservation, contextual fidelity, and semantic resonance. This renders the approach a unique contribution to the emerging field of AI-based memorialization. From both an ethical and legal standpoint, further roll-out must be managed by healthcare and psychology ethics committees, based on guidelines such as the GDPR and the EU AI Act. D.M.A.I.H. presents a novel generative pipeline that combines identity preservation, contextual fidelity, and semantic

resonance, in contrast to current digital commemoration services that are static or symbolic. Because of this, the proposed approach offers something special to the developing field of AI-based memorialization. Based on rules like the GDPR and the EU AI Act, ethics committees in the fields of psychology and healthcare must oversee any future roll-out from a moral and legal perspective.

REFERENCES

[1] B. Acim, H. Ouhnni, N. Kharmoum, S. Ziti, and A. Ouajdouni, "The era of blockchains, deepfakes, and NFTs in therapeutic treatments: A comparative study," in Non-Fungible Tokens (NFTs) in Smart Cities: Advancements and Security Challenges, U. Khalil, M. Uddin, O. Malik, and A. Dandoush, Eds. Hershey, PA, USA: IGI Global Scientific Publishing, 2026, pp. 119–152

[2] W. Abbaoui, S. Retal, N. Kharmoum, and S. Ziti, "Artificial intelligence at the service of precision medicine," in Proc. Int. Conf. Advanced Intelligent Systems for Sustainable Development (AI2SD), Lecture Notes in Networks and Systems, vol. 713, J. Kacprzyk, M. Ezziyyani, and V. E. Balas, Eds. Cham, Switzerland: Springer, 2023.

[3] H. Ouhnni, B. Acim, M. Belhiah, K. El Bouchti, Y. Z. Seghroucheni, S. N. Lagmiri, R. Benachir, and S. Ziti, "The evolution of virtual identity: A systematic review of avatar customization technologies and their behavioral effects in VR environments," Front. Virtual Real., vol. 6, 2025.

[4] B. Acim, N. Kharmoum, M. Ezziyyani, and S. Ziti, "Mental health therapy: A comparative study of generative AI and deepfake technology," in Proc. Int. Conf. Adv. Intell. Syst. Sustain. Dev. (AI2SD 2024), vol. 1403, Lecture Notes in Networks and Systems, M. Ezziyyani, J. Kacprzyk, and V. E. Balas, Eds. Cham, Switzerland: Springer, 2025, pp. 351–357.

[5] B. Acim, N. Kharmoum, and S. Ziti, "Generative AI in medical education: Personalized learning with deepfakes, LLMs, and RAG," in The Second International Symposium on Generative AI and Education (ISGAIE'2025), R. González Vallejo, G. Moukhliss, E. Schaeffer, and V. Paliktzoglou, Eds., Lecture Notes on Data Engineering and Communications Technologies, vol. 262. Cham: Springer, 2025, pp. 175 188.

[6] K. K. Fitzpatrick, A. Darcy, and M. Vierhile, "Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): A randomized controlled trial," *JMIR Mental Health*, vol. 4, no. 2, 2017.

[7] T. W. Bickmore and R. W. Picard, "Constructing and maintaining long-term human–computer relationships," *ACM Transactions on Computer-Human Interaction*, vol. 12, no. 2, pp. 293–327, 2005.

[8] W. Abbaoui, S. Retal, B. El Bhiri, N. Kharmoum, and S. Ziti, "Towards revolutionizing precision healthcare: A systematic literature review of artificial intelligence methods in precision medicine," Informatics in Medicine Unlocked, vol. 46, 101475, 2024.

[9] H. Ouhnni, M. Belhiah, **S. Ziti**, K. El Bouchti, and S. Lagmiri, "Digital afterlife: Challenges and technological innovations in pursuit of immortality," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 37, no. 2, Feb. 2025.

[10] J. R. Brubaker, G. R. Hayes, and P. Dourish, "Beyond the grave: Facebook as a site for the expansion of death and mourning," *The Information Society*, vol. 29, no. 3, pp. 152–163, 2013.

[11] M. Gibbs, J. Meese, M. Arnold, B. Nansen, and M. Carter, "#Funeral and Instagram: Death, social media, and platform vernacular," *Information, Communication & Society*, vol. 18, no. 3, pp. 255–268, 2015.

[12] H. Ouhnni, B. Acim, M. Belhiah, R. Benachir, and S. Ziti, "A rule-based framework for clothing fit recommendation from 3D body 11 | P a g e www.thesai.org (IJACSA) International Journal of Advanced Computer Science and Applications Vol. XXX, No. XXX, 2024 reconstruction," Int. J. Adv. Comput. Sci. Appl. (IJACSA), vol. 16, no. 6, 2025.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., "Generative adversarial Networks," *Advances in Neural Information Processing Systems (NeurIPS)*, Montreal, Canada, 2014.

[14] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019.

[15] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 2022.

[16] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "DreamBooth: Fine-tuning text-to-image diffusion models for subject-driven generation," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada, 2023.

[17] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," *arXiv:2106.09685*, 2021.

[18] J. Ma, J. Gu, Y. Jiang, Y. Zhang, and J. Zhu, "DreamArtist: Towards controllable one-shot text-to-image generation via contrastive prompt-tuning," *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2299–2310, 2023.

[19] S. Walter, "Digital death: Mortality and beyond in the online age," *Mortality*, vol. 20, no. 3, pp. 259–277, 2015.

[20] M. Gibbs, J. Meese, M. Arnold, B. Nansen, and M. Carter, "#Funeral and Instagram: Death, social media, and platform vernacular," *Information, Communication & Society*, vol. 18, no. 3, pp. 255–268, 2015.

[21] K. K. Fitzpatrick, A. Darcy, and M. Vierhile, "Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): A randomized controlled trial," *JMIR Mental Health*, vol. 4, no. 2, 2017.

[22] T. W. Bickmore and R. W. Picard, "Establishing and maintaining long-term human–computer relationships," *ACM Trans. Comput.-Hum.* Interact., vol. 12, no. 2, pp. 293–327, 2005.

[23] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019.

[24] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," arXiv:2302.05543, 2023.

[25] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," arXiv:2103.00020, 2021.

[26] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. Conf. of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), Minneapolis, MN, USA, 2019.

[27] A. S. Luccioni, C. Akiki, M. Mitchell et al., "Stable Bias: Analyzing Societal Representations in Diffusion Models," arXiv:2303.11408, 2023.

[28] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large margin cosine loss for deep face recognition," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5265–5274, 2018.

[29] N. Kumari, B. Zhang, E. Shechtman, and J. Y. Zhu, "Multi-concept customization of text-to-image diffusion," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1931–1941, 2023.

[30] L. Xu, H. Xie, S.-Z. J. Qin, X. Tao, and F. L. Wang, "Parameter-Efficient Fine-Tuning Methods for Pretrained Language Models: A Critical Review and Assessment," arXiv:2312.12148, 2023.

[31] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, "An image is worth one word: Personalizing text-to-image generation using textual inversion," ACM Transactions on Graphics (TOG), Proc. of SIGGRAPH 2023, vol. 42, no. 4, pp. 1–11, 2023.

[32] Y. Tewel, R. Gal, G. Chechik, et al., "Key-Locked Rank One Editing for Text-to-Image Personalization," *Proceedings of SIGGRAPH 2023*.

[33] T. Cilloni, C. Fleming et al., "Privacy Threats in Stable Diffusion Models," arXiv:2311.09355, 2023.

[34] K. K. Fitzpatrick, A. Darcy, and M. Vierhile, "Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (Woebot): A randomized controlled trial," *JMIR Mental Health*, vol. 4, no. 2, 2017

[35] T. W. Bickmore and R. W. Picard, "Establishing and maintaining long-term human–computer relationships," *ACM Transactions on Computer-Human Interaction*, vol. 12, no. 2, pp. 293–327, 2005.

[36] J. R. Brubaker, G. R. Hayes, and P. Dourish, "Beyond the grave: Facebook as a site for the expansion of death and mourning," *The Information Society*, vol. 29, no. 3, pp. 152–163, 2013

Re-engineering Grid-Based Quorum Replication into Binary Vote Assignment on Cloud: A Scalable Approach for Strong Consistency in Cloud Databases

Ainul Azila Che Fauzi[1], Noor Ashafiqa[2], Asiah Mat[3], Syerina Azlin Md Nasir[4], A. Noraziah[5]
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Kelantan, Machang, Kelantan [1, 2, 3, 4]
Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah, Pahang, Malaysia[5]

***Abstract*—The growth of cloud computing has heightened the demand for replication strategies that ensure strong consistency, high availability, and low communication cost across distributed infrastructures. Existing systems such as DynamoDB, FoundationDB, and GeoGauss illustrate different design trade-offs but face limitations in balancing latency, correctness, and resilience under dynamic workloads. This study proposes the Binary Vote Assignment in Cloud (BVAC), a cloud-native replication algorithm re-engineered from the Binary Vote Assignment on Grid Quorum (BVAGQ). BVAC organizes replicas in a logical grid structure and employs binary voting weights with a Commit Coordination (BCC) mechanism to enforce quorum-validated commits, representing a form of quorum-based replication. This design maintains serializable consistency, minimizes replication conflicts, and reduces low communication cost through fixed-size quorums of three to five replicas. Experimental results demonstrate that BVAC sustains fault tolerance, achieves cloud database replication efficiency, and sustains high data availability via multiple valid quorum paths. By avoiding the heavy coordination cost and infrastructure footprint of current systems, BVAC provides a scalable and cost-efficient replication strategy tailored for modern cloud workloads. The study establishes BVAC as an advancement in distributed data management and a foundation for future adaptive and multi-cloud replication frameworks.**

Keywords*—Binary Vote Assignment in Cloud (BVAC); cloud database replication; fault tolerance; high availability; quorum-based replication; strong consistency

I. Introduction

Distributed computing environments present significant challenges, particularly in maintaining high availability and reliability of data. Data replication has long been recognized as a fundamental technique to address these challenges, as it enables systems to provide fault tolerance, load balancing, and continuous access to critical information [1, 2, 3, 4].

In traditional grid computing environments, data replication was introduced to enhance availability and dependability by distributing replicas across multiple grid nodes. This approach allowed systems to tolerate node failures, support concurrent access, and reduce the risk of data loss in large-scale scientific and computational applications [5]. Techniques such as static replication and dynamic replication policies were widely deployed to ensure that frequently accessed data remained available across geographically dispersed resources [6].

While grid-based replication strategies achieved significant success in improving fault tolerance and data accessibility, they also exhibited several limitations. Grid environments are inherently tightly coupled and resource-constrained, which often restricts scalability and adaptability. Furthermore, replication overhead was difficult to manage, and consistency across replicas was not always guaranteed due to network heterogeneity and limited elasticity of grid resources [7]. These challenges made grid-based replication less suitable for modern workloads that demand real-time responsiveness and global accessibility.

Several replication strategies have been proposed to address the challenges of managing data in cloud environments. Broadly, these methods can be categorized into synchronous replication and asynchronous replication. Synchronous replication ensures strong consistency by updating all replicas simultaneously, which is suitable for mission-critical applications but often introduces high latency, especially in geographically distributed clouds [5]. On the other hand, asynchronous replication reduces latency by allowing updates to propagate to replicas at a later stage, thereby improving performance but at the cost of potential data inconsistency during failures [6].

In addition to these traditional methods, several hybrid approaches have been developed to strike a balance between consistency and performance. For example, techniques such as quorum-based replication and majority consensus protocols allow systems to achieve partial synchrony, ensuring data reliability without incurring the full latency penalty of synchronous models [7]. Multi-cloud replication frameworks have also been explored to enhance disaster recovery and reduce the risk of vendor lock-in, but these often come with additional management complexity and resource overhead [8, 9].

Despite these advances, current replication approaches remain limited in handling the highly elastic and dynamic nature of modern cloud workloads [2, 10]. In practice, cloud environments introduce additional challenges such as multi-tenancy, where multiple independent users share the same infrastructure, often leading to unpredictable performance interference. Moreover, geo distribution of cloud datacenters creates latency asymmetries and raises the difficulty of maintaining consistency across regions. Cost-efficiency also becomes a critical factor, as replication policies directly affect storage, bandwidth, and operational costs in pay-as-you-go

models. Traditional replication mechanisms often fail to adapt to these requirements, as they were not originally designed to optimize for resource elasticity, cost sensitivity, and tenant isolation. This gap underscores the importance of designing new replication strategies that can simultaneously ensure high availability, strong consistency, and performance while remaining cost-aware and adaptable to real-time workload dynamics.

With the rise of cloud computing, replication challenges have become more complex. Cloud infrastructures demand solutions that can handle multi-tenant environments and flexible resource allocation while maintaining low latency and strong consistency [7, 10]. Existing replication mechanisms do not adequately address these requirements, leaving a gap for innovative approaches that balance consistency, availability, and performance in large-scale, globally distributed cloud systems.

To address these challenges, this study proposes the Binary Vote Assignment on Cloud (BVAC) algorithm, by re-engineering BVAGQ [11] for cloud environments. BVAC is designed to improve data consistency and availability by leveraging a binary voting mechanism in quorum-based replication, where replica servers are assigned voting weights to determine the validity of transactions. This voting scheme minimizes replication conflicts, enhances fault tolerance, and reduces the communication overhead typically associated with quorum-based replication. Unlike previous grid implementations, the proposed BVAC adapts its design to the elasticity and scalability of cloud infrastructures, enabling efficient replica placement across geographically distributed nodes. By re-engineering BVAGQ for cloud environments, this research aims to bridge the gap between traditional quorum-based replication in grids and the dynamic requirements of modern cloud systems. Unlike existing approaches, the proposed BVAC explicitly addresses multi-tenancy, geo-distributed deployments, and cost-efficiency by incorporating a lightweight voting mechanism that adapts to workload variability while minimizing communication overhead. This makes BVAC particularly well-suited for cloud infrastructures where scalability, elasticity, and operational costs are as critical as data consistency and availability.

The remainder of this paper is organized as follows. Section II presents the literature review on replication strategies in grid and cloud environments. Section III describes the design of the proposed Binary Vote Assignment on Cloud (BVAC) algorithm, including its voting scheme and replica placement strategy. Section IV presents the experimental setup and evaluation methodology, and discusses the results in terms of consistency, availability, and performance. Finally, Section V concludes the paper and outlines directions for future research.

II. Literature Review

Data replication in cloud environments integrates diverse sub-strategies, techniques, and algorithms into cohesive frameworks that sustain system dependability. The replication process is commonly structured around three fundamental phases: identifying frequently accessed data, determining the number of replicas, and selecting their optimal placement. Extensive research has introduced algorithms targeting each of these phases to enhance performance, consistency, and fault tolerance [12, 13]. Modern cloud platforms translate these conceptual strategies into operational practice through protocol-level implementations and system design choices. DynamoDB, for instance, represents a production-grade system in which phase-level replication decisions are directly embedded within concrete protocol mechanisms, thereby exemplifying the alignment of theoretical models with practical deployment.

DynamoDB integrates a tunable-consistency read model into its replication design, providing eventual reads by default for latency reduction and permitting per-request strong reads within a region under ACID semantics with serializable isolation [14, 15]. In multi-region deployments, Global Tables employ asynchronous propagation, which yields eventual cross-region consistency while decoupling regions during inter-region disturbances [15]. The communication path per write remains bounded, as a leader disseminates the update to followers across three (3) availability zones and commits upon a two-of-three quorum, so the critical path corresponds to one intra-region round-trip time with message complexity linear in the replica count, $O(n)$ for $n = 3$. Read placement adheres to the consistency objective, whereby strong reads are directed to the leader and eventual reads may be served by any replica; transactional operations introduce additional coordination rounds via two-phase commit, and any synchronous cross-region configuration, when enabled, elevates both the critical-path round-trip time and the message fan-in. With respect to availability, the three-replica topology tolerates a single replica failure per partition and supports rapid leader re-election, while asynchronously replicated Global Tables allow regional autonomy under wide-area impairments at the expense of temporary divergence; conversely, synchronous multi-region modes trade some write availability to secure stronger cross-region guarantees [15].

Alongside DynamoDB, two further systems illustrate alternative trade-offs, namely FoundationDB and GeoGauss. FoundationDB advances strict serializability across the keyspace through an unbundled control and storage plane comprising commit proxies, resolvers, and log servers, with synchronous replication in a primary region and frequent inclusion of satellite replicas as well as an asynchronous secondary region for disaster recovery [16]. The communication path per writes traverses coordination services and multiple log replicas, often including satellites, which introduces additional hops relative to a three (3) replica quorum and can elevate tail latency under load or under cross-region safety requirements. Steady-state reads obtain a read version from the primary path and avoid extra wide-area round trips. With respect to availability, the architecture tolerates multiple component failures inside a region and supports automatic failover to a secondary region upon primary loss, delivering strong reliability at the expense of additional coordination and replica footprint [16].

GeoGauss pursues strong global consistency for geo-distributed SQL through a full-replica, multi-master architecture that combines epoch-driven optimistic coordination with Raft-style membership, enabling write origination in any region while enforcing a global commit order [17]. The communication cost exceeds single-region designs, since each transaction issues cross-region control traffic for epoch advancement and

validation and ships data to full replicas, which embeds wide-area round-trip time and extra coordination in the commit path, particularly under contention or reconfiguration. Regarding availability, complete replicas across regions furnish resilience and locality for reads and writes, and service can continue through regional failures, while long-haul coordination under partitions or congestion lengthens commit time until quorum and ordering guarantees are satisfied.

A. Comparative Analysis of Cloud Replication Systems

Despite considerable progress, DynamoDB, FoundationDB, and GeoGauss continue to reveal structural limitations along the consistency–latency–availability frontier. DynamoDB's configurable consistency model achieves low latency but risks stale reads; strong reads are restricted to regional scope, and globally strong modes require wide-area coordination. Its otherwise lightweight two-of-three quorum inflates latency under synchronous multi-region deployment, while the three-replica design tolerates only a single failure, reducing write availability during WAN impairments [14]. FoundationDB delivers strict serializability yet relies on a multi-role control path involving commit proxies, resolvers, and log servers. This design increases coordination sensitivity and tail latency under load. Although failover is robust, sustaining cross-region safety typically demands five or more replicas, thereby elevating resource cost [16]. GeoGauss ensures strong global consistency through full replicas and epoch-based OCC ordering under Raft membership, but wide-area validation and full data movement in the commit path increase coordination overhead. While regional failures can be absorbed, partitions delay progress until global ordering and quorum are re-established, imposing the heaviest infrastructure burden due to per-region full replicas [17].

In our previous studies, BVAGQ found that write-query availability might be enhanced with minimal communication costs by employing a limited replication quorum [18]. Nonetheless, that methodology did not encompass cloud-native environments. This paper introduces the Binary Vote Assignment in Cloud (BVAC), a re-engineered variant designed to support synchronous replication across fragmented databases, with BVAGQ providing disjoint partitioning of data. BVAC enforces strong-by-default consistency by validating commits through vote-based quorums of constant size, thereby preserving serializability without incurring continuous wide-area coordination. Communication overhead is bounded by a fixed quorum of three to five servers, which reduces message complexity and constrains commit round-trip latency. Data availability is sustained through a binary vote assignment that admits multiple valid quorum paths, ensuring resilience under failures. In this way, BVAC retains the lightweight advantages of BVAGQ while extending its applicability to cloud infrastructures with explicit support for fragmentation and cross-region fault tolerance. The primary strategies for database fragmentation are horizontal fragmentation and vertical fragmentation. Horizontal fragmentation grants users' access to all attributes. Vertical fragmentation partitions the database based on properties, rather than entire rows [19, 20, 21] (Table I).

TABLE I. THIS SECTION ANALYSES EXISTING CLOUD DATABASE SYSTEMS

System Method	Data Consistency	Commit Path	Communication Cost
Dynamo DB	Configurable consistency model: default eventual consistency, optional strong consistency per request, with ACID serializable transactions.	- Primary replica handles write with 2-of-3 quorum - Strong reads from primary, eventual reads from any secondary. - Transactions require 2PC. - Async global tables reduce cost, while synchronous multi-region adds cross-region latency and coordination.	3 replicas per partition (regional); extra regions for global tables.
Foundation DB	Strict serializability through synchronous replication across primary and secondary roles.	- Strict serializability via synchronous replication. - Writes coordinated through commit proxies and multiple log servers (primary + secondaries), - Strong correctness and failover support, but higher coordination overhead and resource cost.	Often ≥5 replicas incl. satellites/coordinators (configurable).
GeoGauss	Strong global consistency through full-replica multi-primary with epoch-based OCC.	- Multi-primary architecture with full replicas in all regions. - Any primary can accept writes, but global commit order requires cross-region coordination. - Strong global consistency and resilience.	Full replica per region - high communication cost and cost overhead.

III. METHODOLOGY

The fundamental concept of replication is creating numerous copies of identical data or replicas across various storage locations. This research introduces the Binary Vote Assignment in Cloud (BVAC) method. In BVAC, all servers are systematically arranged in a two-dimensional grid configuration. If BVAC comprises twenty-five servers, they will be systematically arranged in a 5 x 5 grid format.

A. BVAC Algorithm Definition

In this section, the Binary Vote Assignment in Cloud (BVAC) is proposed by considering the distributed database fragmentation. The following notations are defined:

i. C is a table within the database.
ii. c' is the instance in C and C'
iii. $T(C)^1$ is the four servers in the corners
iv. $T(C)^2$ is the alternative locations on the peripheries
v. $T(C)^3$ is the central locations
vi. T is a transaction.
vii. x is a variable in C that gets altered by element of T.
viii. y is a variable in C that expected to remain unaltered by element of T.
ix. C_1 is a vertical fragmented table with data x.
x. C_2 is a horizontal fragmented table with data x.
xi. Pk is a primary key.
xii. Pk,x is a primary key with data x.
xiii. Pk,y is a primary with data y, where $y \neq x$

xiv. $C_{1_{(Pk,x)}}$ and $C_{1_{(Pk,y)}}$ are a horizontal fragmentation relation.

xv. η and ψ are groups for the transaction T.

xvi. $\lambda = \eta$ or ψ where it represents different transaction T (before and until get quorum).

xvii. T_η is a set of transactions that comes before T_ψ, while T_ψ is a set of transactions that comes after T_η.

xviii. D is a union of all data objects managed by all transactions T of BVAC.

xix. BVAC transaction element T_λ is an element either in different set of transactions T_η or T_ψ.

xx. wT_λ is write counter for the transaction.

xxi. $\hat{V}_{\lambda_x}$ is a transaction that is transformed from T_{λ_x}

xxii. T_{μ_x} represents the transaction feedback from a neighbour site.

xxiii. T_{μ_x} exists if either T_{λ_x} or $\hat{V}_{\lambda_x}$ exists.

xxiv. Successful transaction at primary site $T_{\lambda_x} = 0$ where $T_{\lambda_x} \in D$ (i.e., the transaction locked an instant x at primary). Meanwhile, successful transaction at neighbour site $T(\mu_x) = 0$, where $\mu_x \in D$ (i.e., the transaction locked a data x at neighbour).

xxv. $\left\lceil \frac{n}{2} \right\rceil$ is the greatest integer function (i.e., n=9, $\left\lceil \frac{9}{2} \right\rceil = 5$.

B. Data Replication in BVAC

The Algorithm 1 details the replication process for BVAC via BVAC Commit Coordination (BCC), showing how a commit initiated at a primary and a neighbour server is executed.

Algorithm 1: The BCC Algorithm: Data Replication

```
1  manage_bvaqgar_transaction ( )
2  {
3  while (InComplete)
4  do
5    while (transStat ≠ "Abort")
6    do
7          /* receive T_λx || V̂_λx || T_μx where λ = η, ψ either
       from client or any BTM of replica i ∈ T(C)*/
8      receive (client @ BCC of i ∈ T(C): T_λx || V̂_λx);
9      pid_T_λx = process id of T_λx;
10     log_tT_λx = login time of T_λx;
11          /*recognize replica task either to be as primary or
            neighbour processing for T_λx, λ = η, ψ */
12     switch {
13      case (receive(client @ BCC of neighbour: T_λx || V̂_λx||
       T_μx):
14       primary_replica_processing ( );
15                  break;
16              case (receive(BCC of primary: T_λx || V̂_λx)):
17                  neighbour_replica_processing ( );
18       break;
19        }
20     receive (BCC: V̂_λx, x$erv_Vote || V̂_λx, x$erv_Vote,
21      $_LCount,uT_λx);
22     if (x$erv_Vote = 1) then
23              Commit V̂_λx ;
24     endif
25     if (x$erv_Vote = 0);
26     Endif
27   /*On receiving transStat = "Abort" from other replica and
     needs to release its lock*/
28   if (receive (BCC: V̂_λx , ∈ T_η transStat, PrimaryID)) then
29     T_λx ∈ V_ψ =T_λx ; /*current T_λx becomeT_λx ∈ T_ψ */
30     V̂_λx =V̂_λx ∈ T_η ; /*V̂_λx ∈ T_η that BCC received will
     survive*/
31   Abort T_λx ;
32     Rollback;
33          T_λx = 1; /*Target Set is equal to 1, means primary
     already gets lock*/
34   endif
```

Every server comprises a primary data file. A server is either functional or nonfunctional, and the status (functional or nonfunctional) of each site is significantly uncorrelated with the others. When a site is functional, the data at the server is accessible; otherwise, it is inaccessible.

For example, in Fig. 1, data from site A (a) is duplicated to its adjacent servers B and D. Consequently, site A possesses three replica servers. Site E has four adjacent servers, which are B, D, F, and H. Therefore, site E possesses five replica servers. Simultaneously, data from site F is duplicated to servers C, E, and I, indicating that server F possesses four clones.

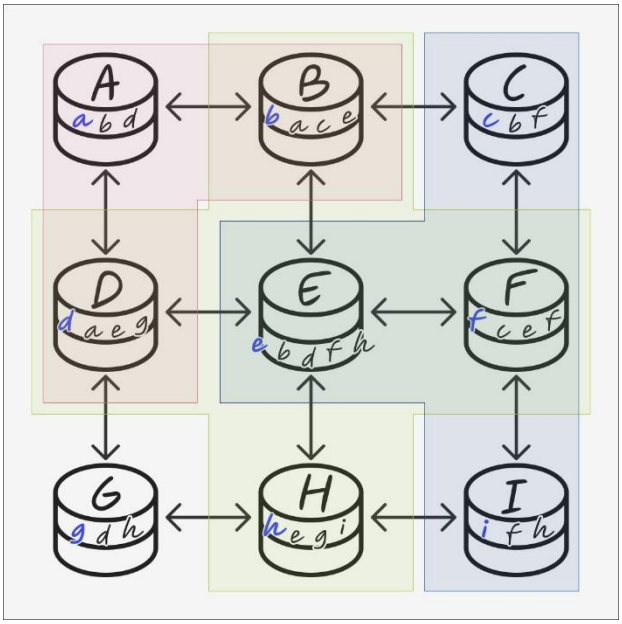


Fig. 1. BVAC framework.

The primary replica at a specific data x is the replica that acknowledges the client's request. In the BVAC algorithm, each replica of *T(C)* can simultaneously function as both a primary and a neighbor replica. Any replica $i \in T(C)$ may be designated as the primary replica, whereas additional replicas, $j \in T(C)$ where $i \neq j$ are considered neighbours. Line 7 processes a transaction T_{λ_x} that requests an update of data x from any replica of *T(C)*. Line 8-18 establish that the replica will serve as the primary, while the others will function as neighboring replicas to execute T_{λ_x}.

Recall our previous work, let *T(C)* be the set of replicas with replicated copies are stored corresponding to the assignment *C* for particular instant *x*, $T(C_x) = \{m(i,j), m(i-1,j), m(i,j-1), m(i,j+1), m(i+1,j)\}$. Two sets of transactions, T_η request instant *x* from *m(i,j)* replica, while T_ψ request instant *x* from *m(i-1,j)* respectively. The *m(i,j)* replica functions as the primary replica for processing T_η, where *m(i-1,j), m(i,j-1), m(i,j+1), m(i+1,j)* are neighbour replicas for processing $T_{\gamma_x} \in T_\eta$. Simultaneously, *m(i-1,j)* replica functions as the primary replica for processing T_ψ, while *m(i,j-1), m(i,j+1), m(i+1,j)* and *m(i,j)* are neighbour replicas for processing $V_{\gamma_x} \in T_\psi$. Both *m(i,j) and m(i-1,j)* replicas execute two different processing tasks concurrently. The *m(i,j)* replica is the primary replica managing T_η and its adjacent replica managing T_ψ, whereas the *m(i-1,j)* replica is the primary replica for processing T_ψ and neighbour replica for processing T_η. BVAC model considers different sets of transactions T_η and T_ψ. T_η is a set of transactions that comes before T_ψ, while T_ψ is a set of transactions that comes after T_η. The effect of BVAC transaction is defined as the processing of one instance of the transaction.

IV. Experimental Results

A. BVAC Algorithm Definition

To illustrate the operation of commit coordination under the BVAC Commit Coordination (BCC) mechanism, consider the case where two distinct transaction sets, T_η and T_ψ, concurrently request access to data file e at replicas E and B, respectively.

A cluster of five replication servers, interconnected as shown in Fig. 2, is used to illustrate the BVAC mechanism. Each primary replica propagates its database state to neighbouring replicas, allowing clients to access data from any server holding a replica. Consider two distinct transaction sets, T_η and T_ψ both requesting access to data file e at replicas E and B, respectively. When T_η and T_ψ attempt to update *e*, they must first issue update requests to their respective primary replicas, B and E. Both T_η and T_ψ propagate lock requests, but only the first transaction to acquire the lock proceeds, while the other is aborted. Consequently, replicas B and E maintain pending transactions, yet neither can read or update e concurrently. Primary node E, $T_{\eta,e} = 1$ propagates lock requests to neighbours B, D, F, and H, while primary node B $T_{\psi,e}$ propagates locks to neighbours E, D, F, and H. The transaction that first secures a majority quorum is transformed into $\hat{V}_{\lambda_x} \in V_\eta, V_\psi$. The details of experimental outcomes are summarized in Table II.

Assertion: If the transaction gets all locks from replica $i \in T(C)$, then the transactions will be executed successfully.

Proof: The only way that a transaction gets a lock in initiate lock is when $T_{\lambda_x} = 1$ with $T_{\lambda_x} \in T_\eta$. After $T_{\lambda_x} \in T_\eta$ success to initiate lock at a server, then, $T_{\lambda+1_x}$, ..., $T_{\lambda+q_x}$ which are the elements that exist in T_η will be queued. To get majority quorum, $wT_{\lambda_x} \geq \left\lceil \frac{n}{2} \right\rceil$ is required. This means that the primary server needs to get the majority locks from its neighbour servers by calling request lock from the neighbours servers. Each neigbours $i \in T(C)$ will send feedback to the primary to notify it is in free lock or not. If the primary gets the majority locks of instant *x*, it means that $\hat{V}_{\lambda_x} = T_{\mu_x} = T_\eta$ where $\hat{V}_{\lambda_x}$ gets a quorum. Next, the primary will send error notification to other neighbours $i \in T(C)$ in the quorum. Consequently, when $T_{\lambda_x} \in T_\psi$ releases its lock, $T_{\lambda_x} \in T_\eta$ gets the lock from every neighbour $i \in T(C_x)$. After T_{λ_x} gets majority quorum, relation *T* is fragmented into T_1 and T_2 using vertical fragmentation. Again, T_1 is fragmented into $T_{1_{(Pk,x)}}$ and $T_{1_{(Pk,y)}}$ using horizontal fragmentation. When a user finishes updating the instant, $\hat{V}_{\lambda_x}$ commit (send the fragmented data) to ∀ neighbour $i \in T(C_x)$. Therefore, all replicas of $T(C_x)$ will perform and execute the update successfully.

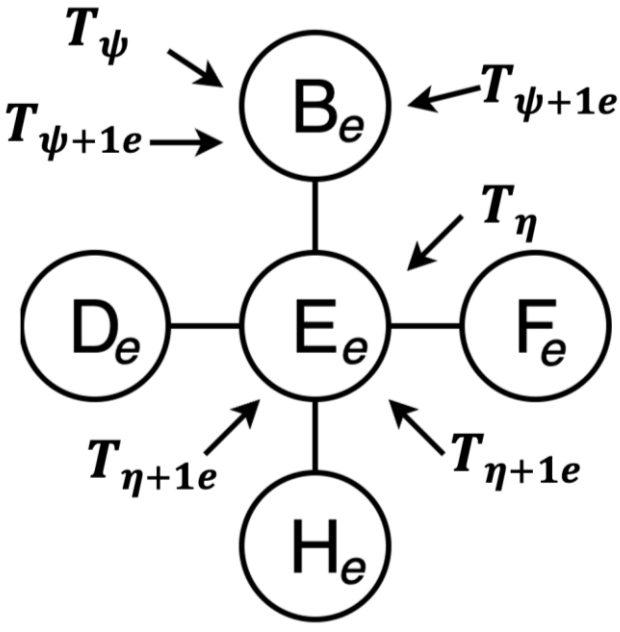


Fig. 2. An example of BVAC transaction requests.

TABLE II. AN EXAMPLE OF HOW BVAC HANDLE CONCURRENT TRANSACTIONS

REPLICA	E	B	D	F	H
TIME					
t1	Unlock (e)	unlock (e)	unlock (e)	unlock (e)	unlock (e)
t2	Request update				
t3	write l(e) counter_w(e)=1	Request update			
t4	wait	write lock(e) counter_w(e)=1			
t5	propagate lock:B				
t6	propagate lock:D	propagate lock:D			
t7			lock(e) from E		
t8	get lock:7 counter_w(e)=2	propagate lock:F			
t9	propagate lock:F			lock(e) from B	
t10	propagate lock:H	get lock:F counter_w(e)=2			
t11		propagate lock:H			lock(e) from E
t12	get lock:H counter_w(e)=3	propagate lock:E			
t13	obtain quorum release lock:3	propagate lock:H			
t14		abort $T_{\lambda_e} \in T_{\psi}$ & rollback, lock(e) from 8		rollback, lock(e) from E	
t15	update e				
t16	fragment *T' And T'$_1$*				
t17	commit $\hat{V}_{\lambda_e} \in V_{\eta}$	commit $\hat{V}_{\lambda_e} \in V_{\eta}$	commit $\hat{V}_{\lambda_e} \in V_{\eta}$	commit $\hat{V}_{\lambda_e} \in V_{\eta}$	commit $\hat{V}_{\lambda_e} \in V_{\eta}$
t18	Unlock (e)	Unlock (e)	Unlock (e)	Unlock (e)	Unlock (e)

B. Communication Cost Comparison

In this section, we compare the storage usage of BVAC, DynamoDB, FoundationDB, and GeoGauss. In BVAC, replication is bound to a fixed quorum size of three to five servers. This allows transactions to commit using a constant quorum, which limits the storage overhead even when the number of servers increases. In DynamoDB, each partition is synchronously replicated across three replicas within a region. When Global Tables are enabled, data is further replicated asynchronously across multiple regions, meaning the storage requirement increases proportionally with the number of regions. FoundationDB, by contrast, relies on a more complex configuration involving commit proxies, resolvers, and log servers. To ensure cross-region durability and strict serializability, the system often requires five or more replicas, including satellites, which elevates the storage footprint. GeoGauss employs the most storage-intensive approach, as it maintains a full replica of the database in every region. This design provides strong global consistency and resilience but comes at the cost of replicating the entire dataset across all regions. Table III shows the comparison between BVAC, DynamoDB, FoundationDB, and GeoGauss in terms of storage use.

From Table III, it is apparent that BVAC requires the least storage overhead by restricting replication to a small, constant quorum size of three to five servers. DynamoDB demands three replicas per partition in each region, and the storage requirement expands with the number of regions deployed in Global Tables. FoundationDB requires at least five replicas to preserve strict serializability and cross-region fault tolerance, thereby consuming more resources. GeoGauss incurs the heaviest storage cost since every region maintains a complete replica of the database. Consequently, BVAC demonstrates superior storage efficiency compared with DynamoDB, FoundationDB, and GeoGauss, making it a practical option for large-scale cloud systems where both consistency and resource optimization are critical.

TABLE III. STORAGE USAGE COMPARISON OF BVAC, DYNAMODB, FOUNDATIONDB, AND GEOGAUSS

Replication Techniques	Number of Servers	Storage Use
BVAC	3–5 replicas (constant quorum)	3–5
DynamoDB	3 replicas per partition (regional); extra regions for global tables	3 per region, grows with Global Tables
FoundationDB	≥5 replicas (primary, secondaries, satellites)	5 or more
GeoGauss	Full replica in every region	Equal to number of regions (high cost)

V. CONCLUSION

The rapid evolution of cloud computing has increased the significance of replication systems that can concurrently guarantee consistency, availability, and efficiency at scale. Conventional grid-based methodologies and commercial cloud platforms have advanced replication practices, but they continue to face inherent limitations within the consistency, latency and availability spectrum. This research presents the Binary Vote Assignment in Cloud (BVAC) as a cloud-native advancement of the BVAGQ replication model to tackle these ongoing challenges. BVAC integrates binary vote assignment into a quorum-based structure and utilizes the BVAC Commit Coordination (BCC) mechanism to authenticate transactions on fixed-size quorums across distributed replicas. This approach maintains robust consistency, minimizes communication complexity, ensures reliable fault tolerance with low latency, and improves storage efficiency by requiring fewer replicas than existing systems. Collectively, these contributions demonstrate BVAC as a scalable, reliable, and economical replication framework for extensive cloud settings. While the results of this study are promising, several aspects warrant further investigation. The current evaluation of BVAC is limited to specific workloads; its scalability and adaptability under highly dynamic and heterogeneous cloud conditions remain to be explored. In particular, BVAC could be extended with adaptive quorum resizing and workload-aware strategies to dynamically respond to shifting read/write ratios and latency requirements. Furthermore, future research should consider strengthening security and trust mechanisms, such as integrating Byzantine fault tolerance or lightweight blockchain-inspired auditing, to protect against adversarial conditions like malicious voting or quorum manipulation. Potential trade-offs, such as the complexity of quorum management, deployment across multi-cloud environments, and performance under real-world changeable workloads, need deeper analysis.

ACKNOWLEDGMENT

This research was funded by the Fundamental Research Grant Scheme – Early Career (FRGS-EC), Ministry of Higher Education Malaysia, under Grant Nos. FRGS-EC/1/2024/ICT08/UITM/02/3, 600-RMC/FRGS-EC 5/3 (053/2024). The authors also acknowledge the support of Universiti Malaysia Pahang Al-Sultan Abdullah under Grant No. RDU253002.

REFERENCES

[1] J. Eom, "Efficient data replication for fault tolerance in distributed computing environments," *Int. J. Distrib. Syst. Technol.*, vol. 8, no. 3, pp. 1–15, 2017.

[2] R. Dugyani and D. Govardhan, "Data replication and scheduling in the cloud with optimization assisted workflow management," Web Intelligence, vol. 22, no. 1, pp. 55–72, 2024.

[3] K. Acquah, "Empirical insights into replication models for distributed database environments," Int. J. Comput. Appl. (IJCA), vol. 186, no. 53, pp. 1–7, 2024.

[4] T. Taipalus, "Database management system performance comparisons: A systematic literature review," J. Syst. Softw., vol. 208, pp. 111–123, 2024.

[5] T. Hamrouni, R. Mokadem, and A. Khelifa, "Review on data replication strategies in single vs. interconnected cloud systems: Focus on data correlation-aware strategies," *Concurrency Computat.: Pract. Exper.*, vol. 35, no. 11, pp. e7612, 2023.

[6] A. Tahir, S. U. Khan, and N. Min-Allah, "Dynamic replication strategies in data grid systems: A survey," *Future Gener. Comput. Syst.*, vol. 111, pp. 654–667, 2020.

[7] S. Rehman, M. A. Jan, and S. Khan, "Survey on data replication in cloud systems," *Cluster Comput.*, vol. 25, pp. 1105–1124, 2022.

[8] M. M. Alshammari, A. A. Alwan, A. Nordin, and A. Z. Abualkishik, "Data backup and recovery with a minimum replica plan in a multi-cloud environment," *Procedia Comput. Sci.*, vol. 141, pp. 45–52, 2018.

[9] J. Alonso, R. Garcia-Castro, J. Cubo, et al., "Understanding the challenges and novel architectural models for multi-cloud native applications," *J. Cloud Comput.*, vol. 12, no. 1, pp. 1–22, 2023.

[10] G. Girau, "Towards adaptive replication policies for elastic cloud workloads," *IEEE Access*, vol. 12, pp. 45112–45126, 2024.

[11] A. Noraziah, A. A. C. Fauzi, S. H. S. A. Ubaidillah, B. Alkazemi, and J. B. Odili, "BVAGQ-AR for fragmented database replication management," *IEEE Access*, vol. 9, pp. 56168–56177, 2021, doi: 10.1109/ACCESS.2021.3065944.

[12] M. A. Fazlina, R. Latip, H. Ibrahim, and A. Abdullah, "Replication strategy with comprehensive data center selection method in cloud environments," *Computers, Materials & Continua*, vol. 74, no. 2, pp. 4139–4155, 2023

[13] A. Kaur, P. Gupta, M. Singh and A. Nayyar, "Data placement in era of cloud computing: A survey, taxonomy and open research issues," Scalable Computer, vol. 20, no. 2, pp. 377–398, 2019.

[14] V. Gupta, S. Kharche, A. Lakshman, S. Mittal, and R. Sumbaly, "Amazon DynamoDB: A scalable, predictably performant, and fully managed NoSQL database service," *Proc. USENIX Annu. Tech. Conf. (ATC)*, pp. 511–525, 2022.

[15] J. Idziorek, A. Keyes, C. Lazier, et al., "Distributed transactions at scale in Amazon DynamoDB," Proc. USENIX Annu. Tech. Conf. (USENIX ATC), pp. 675–690, 2023.

[16] Y. J. Zhou, M. Xu, A. Shraer, et al., "FoundationDB: A distributed unbundled transactional key-value store," Proc. ACM SIGMOD Int. Conf. Manag. Data (SIGMOD), pp. 2653–2666, 2021.

[17] G. Li, L. Zhou, Z. Zhang, et al., "GeoGauss: Strongly consistent and light-coordinated OLTP for geo-replicated SQL database," *Proc. ACM Manag. Data (PACMMOD)*, pp. 1–25, 2023.

[18] A. Noraziah, A. A. C. Fauzi, W. M. W. Mohd, et al., "Managing MyGRANTS fragmented database using Binary Vote Assignment Grid Quorum with Association Rule (BVAGQ-AR) replication model," Proc. Int. Conf. Data Eng. (DaEng-2015), Lect. Notes Electr. Eng., vol. 520, 2019.

[19] A. A. Che Fauzi, W. F. Wan Abdul Rahman, A. Fauzi, et al., "Managing fragmented database in distributed database environment," *J. Math. Comput. Sci. (JMCS)*, vol. 7, no. 1, pp. 8–14, 2021.

[20] M. Aggarwal, S. B. Bajaj, and V. Jaglan, "Performance analysis of degree of redundancy for replication in distributed database system," Proc. Int. Conf. Informatics (ICI), pp. 176–180, 2022.

[21] M. Goel and S. B. Bajaj, "Comparative analysis of vertical fragmentation techniques in distributed environment," *Int. J. Electr. Electron. Comput. Sci. Eng.*, vol. 5, no. 1, pp. 48–52, Feb. 2018.

Optimized Random Forest for High-Accuracy Autism Spectrum Disorder Detection via Phenotypic Data

Mohamed Gawish, Nada S. El-Askary, Mohamed Mabrouk Morsey,
Abeer M. Mahmoud, Mostafa Aref, Taha Ibrahim El-Arif
Computer Science Department-Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt 11566

***Abstract*—Autism Spectrum Disorder (ASD) is a mental disorder with a neurological condition noticed in patients by their persistent deficits in social communication and interaction, along with the possibility of the presence of repetitive motor behaviors or activities. Early diagnosis of this disorder is crucial for improving patients' cognitive, emotional, and social development. Numerous studies on detecting autism exist; however, data limitations and imbalance affect their model generalization. This study proposes a new intelligent computational model for mental healthcare in individuals with ASD, utilizing machine learning (ML) to address these shortcomings. The proposed model enhances the random forest (RF) algorithm by setting optimal parameters and encompasses two key pipelines: 1) the data pipeline and 2) the learning pipeline. We first gathered a multi-source dataset, implemented integration and preprocessing via ML algorithms. The phenotypic data used were collected from 19 different sites and merged to ensure the diversity of the data used. The resulting dataset is subsequently fed into the learning pipeline, where a supervised ML algorithm is employed to create a trained computational model for detecting ASD. The model is based on tuning the RF algorithm by finding the optimal values for five key hyperparameters. After tuning the model, the accuracy of detecting ASD from phenotypic data reached 96.86%, with a sensitivity of 97.14% and a false positive rate of 3.39%. Comparing the tuned RF model with different ML models verified that tuning and optimizing RF achieves a preeminent classification accuracy for ASD detection using phenotypic data, as the accuracy of RF without tuning is 95.06%. In addition, to validate the tuned RF model's real-world applicability, a separate qualitative study was conducted on five independent, narrative-based case studies, where the model accurately classified four of them by translating descriptive language into quantitative features.**

***Keywords*—Mental healthcare; ASD; phenotypic data; ABIDE-II; random forest; hyperparameter optimizations**

I. Introduction

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder that affects the social communication and interaction between a patient and other people and may result in limited and recurring interests, activities, or behavioral habits. According to the World Health Organization [1], the prevalence of ASD has been steadily rising over the past few decades, affecting approximately 1 in 100 children in the world's population and about 3 in 100 children in the United States alone [2]. The causes for this neurological disorder are still not clearly defined, but many scientific researchers have suggested that it may be caused by genetic or clinical factors. For prompt intervention and better results for those impacted, early identification and diagnosis of ASD are essential.

Traditionally, ASD diagnosis has relied on comprehensive clinical assessments, which involve extensive evaluations by trained professionals, including psychologists, psychiatrists, and speech therapists [3]. However, this process can be time-consuming, costly, and subject to inter-rater variability, leading to delayed diagnosis and intervention. Therefore, there is a growing need for more efficient and accurate methods to detect ASD in children. Recent advancements in machine learning (ML) and data mining techniques have shown promise in enhancing the early detection of ASD. In particular, the utilization of phenotypic data has emerged as a valuable resource for developing predictive models.

Phenotypic data refer to the observable characteristics or traits of individuals with ASD, such as communication and social skills, repetitive behaviors, and sensory sensitivities. It can be collected through various sources, such as parental questionnaires, clinical assessments, electronic health records, direct observation, caregiver reports, and wearable sensors [4]. The breadth of behaviors and traits linked to ASD can be better understood using phenotypic data, which can also be utilized to create interventions that focus on particular symptoms. In the context of systems powered by artificial intelligence (AI), phenotypic data can be used to train ML algorithms to recognize patterns that indicate the presence of autism.

The Autism Brain Imaging Data Exchange (ABIDE) dataset, a publicly available dataset [5], is a significant resource for ASD research. It comprises two large-scale collections, ABIDE-I and ABIDE-II, each providing functional magnetic resonance imaging (fMRI) data along with a corresponding phenotypic dataset. ABIDE-II, collected from over 19 diverse sites, introduced more detailed phenotypic characterization not found in ABIDE-I [6]. This dataset is crucial for our research as it provides a comprehensive view of the phenotypic characteristics of individuals with ASD, which is essential for developing accurate ML models for ASD detection. ABIDE-II's phenotypic dataset comprises 347 columns encompassing demographic, diagnostic status, and an extensive array of behavioural assessments [6]. Additionally, patients' basic information, including sex, age, handedness index score, and handedness category, is utilized [6].

One ML algorithm that has gained popularity in the field of medical data is the Random Forest (RF) algorithm. RF is an ensemble learning technique that generates predictions by combining several decision trees. It has been widely used in

various domains due to its ability to handle high-dimensional data, handle missing values, and provide robust predictions [7]. The use of phenotypic data with ML techniques, particularly RF, holds promise for improving the precision and efficiency of ASD detection. Studies have shown that the nature of phenotypic data can enhance the discriminative power of ML models, enabling them to capture subtle patterns indicative of ASD. By leveraging a diverse array of features derived from behavioral observations, cognitive assessments, and physiological measures, the amalgamation of phenotypic data and RF can contribute to a more comprehensive and accurate diagnostic framework.

This research study presents a novel and promising approach to developing a computational model for mental healthcare, focusing on the early detection of ASD. The proposed model aims to distinguish ASD from typical control (TC) individuals using the ABIDE-II phenotypic dataset. We hypothesize that by leveraging a comprehensive set of tuned hyperparameters for RF along with a set of phenotypic features, we can develop a reliable and accurate RF model for early detection of ASD. The proposed model has the potential to highlight a set of hyperparameters for building RF model to guide researchers in their studies and also using the model by healthcare professionals to assist them in making informed decisions and improving the efficiency of ASD screening processes. The main contributions of this study are as follows:

- We proposed an enhanced integrated framework that is based on the random forest algorithm for optimal parameter setting.
- We employed two pipelines, the first of which seeks to overcome the data limitation issue by integrating more than 19 sources of phenotypic data, and then extensive preprocessing is performed.
- We developed a learning pipeline that seeks to find the optimal parameters and address the model generalization issue.
- We validated the proposed model using narrative-based real-world case studies to provide an aided diagnostic tool in the medical domain.

The article is organized as follows: Section II presents a comprehensive literature review, highlighting the limitations and challenges. Section III outlines the methodology of the proposed model and provides detailed descriptions of the techniques employed in the two proposed pipelines. Section IV presents the experimental results, data integration, data preprocessing and the performance evaluation of the developed model. Section V discusses the results of the developed model and presents a comparative analysis. Section VI presents the predictive capabilities of the model using real-world cases. Section VII concludes the findings and potential future research directions.

II. Literature Review

In the era of AI, numerous researchers have explored the use of ML techniques to detect ASD in children. The researchers provided studies that highlighted the potential of utilizing various types of data, including genetic, neuroimaging- ing, and phenotypic data, to develop models that can predict, detect, and classify ASD. Nada S. et al. [8] published a comprehensive survey of studies that have developed ML models to detect brain neurological disorders. The survey included a list of various ML classifiers with different data types, which achieved varying levels of accuracy. It highlighted that in recent years, the utilization of phenotypic data has gained significant attention in the field of ASD detection due to its ability to deliver comprehensive behavioral and clinical information. Additionally, it addressed the challenges, including the availability of large datasets and the need for proper data preprocessing. Focusing on phenotypic data, here are some noteworthy examples of studies and applications that utilize AI techniques to detect ASD with phenotypic data.

1) Perochon et al. [9] developed and validated a mobile digital behavioral phenotyping application called SenseToKnow. The app utilizes computer vision and ML to analyze children's behavioral responses during pediatric well-child visits. The study involved a cohort of toddlers aged 17-36 months, including those diagnosed with ASD (49 children), developmental delay without autism (98 children), and TC (328 children). Children's behavior was recorded using an iPad, and 23 digital phenotypes were extracted, including gaze patterns, facial expressions, head movements, response to name, blink rate, and motor behaviors. The model was trained using extreme gradient boosting (XGBoost), a method that utilizes 1,000 tree-based classifiers, and was evaluated through five-fold cross-validation. It finally achieved a sensitivity of 87.8%.

2) Mazumdar et al. [10] presented a study on the early detection of ASD in children using an approach that depends on gathering data from tracking the patient's eye and then feeding it to an ML classifier. A total of 24 children (14 ASD and 10 TC) were included in the analysis, which utilized 400 natural images sourced from the Saliency4ASD [11] dataset. Using a TreeBagger (bagged DT) classifier trained on the Saliency4ASD dataset, it achieved an accuracy of 59%.

3) Alkahtani et al. [12] developed a facial landmark-based system using DL and ML algorithms to detect ASD in children aged 2 to 8 years. A total of 2,940 images (1,327 ASD and 1,613 TC) were used for analysis, obtained from Kaggle [13]. The study applied eight ML classifiers to the ASD dataset along with their respective accuracy rates: support vector machine (SVM) (81.46%), linear regression (LR) (82.14%),

4) RF (78.06%), k-nearest neighbor (KNN) (52.38%), decision tree (DT) (66.15%), neural network multi-layer perceptron (MLP) (77.04%), gradient boosting (GBoost) (75.51%), and convolutional neural networks (CNN) (92%).

5) Raj et al. [14] explored the effectiveness of ML and DL models in predicting ASD across three population groups: children, adolescents, and adults. The study utilized three publicly available datasets containing behavioral features and demographic information related to ASD from the UCI

machine learning repository: children (292 cases), adolescents (104 cases), and adults (704 cases). The study applied six ML classifiers on the ASD datasets: SVM, LR, Naıve Bayes (NB), KNN, NN, and CNN. They all achieved classification accuracy ranging from 80.95% to 99.53%. CNN consistently outperformed all other classifiers across datasets, demonstrating higher accuracy and robustness.

6) Akter et al. [15] developed ML models capable of detecting ASD at an early stage by leveraging behavioral data, thereby facilitating early interventions. In addition to the datasets utilized by Raj et al. [14], the authors utilized other publicly available datasets related to ASD from the Kaggle repository and combined them based on population group. The aggregated datasets comprise 2,009 cases (toddlers = 1,054, children = 248, adolescents = 98, and adults = 609) with 21 features. Three feature transformation methods were employed to reduce skewness, normalize the data to a range of -1 to 1, and enhance the performance of the ML model. These methods are Log transformation, Z-score normalization, and Sine transformation. A total of 250 classifiers were applied to these datasets, with 80 demonstrating good performance. Classifiers with accuracies under 70% were excluded. The final selection consists of 9 algorithms: adaboost (AB), flexible discriminant analysis (FDA), DT, boosted generalized linear model (GLMBoost), linear discriminant analysis (LDA), mixture discriminant analysis (MDA), penalized discriminant analysis (PDA), SVM, and classification and regression tree (CART). Optimal classifier per age group: SVM (mean accuracy = 98.77%) for the toddlers' dataset, AB (mean accuracy = 97.20%) for the children dataset, AB (mean accuracy = 98.36%) for the adults' dataset, and GLMboost (mean accuracy = 93.89%) for the adolescents' dataset.

7) Erkan et al. [16] investigated the utility of supervised ML models- kNN (k=3), SVM (radial basis function), and RF (60 trees)—for classifying ASD using three age-specific datasets (children, adolescents, and adults) from the UCI repository with a total of 1,100 samples. Each dataset, comprising 20 features including standardized screening questions and demographic variables, underwent rigorous preprocessing including missing value imputation and numeric encoding. Models were evaluated using repeated stratified train/test splits (a = 50%, 60%, 70%, 80%, 90%). Results demonstrate that RF achieved 100% performance across all metrics and datasets, outperforming SVM (accuracy ranged from 91%–100%) and kNN (Accuracy ranged from 83%–95%). The findings underscore the potential of ML, particularly ensemble-based models, in supporting fast, accurate, and scalable ASD diagnosis across age groups. This work supports the integration of ML into clinical ASD screening protocols, especially where large and well-structured data is available.

8) Using the same datasets as those in Erkan et al. [16], Bawa et al. [17] implemented six supervised learning models—LR, SVM, RF, DT, NB, and KNN. Results showed that LR and SVM performed optimally across binary classification (yes/no for ASD) tasks, achieving 94.30 – 99.00% accuracy in individual groups and achieving 97.20% on merged data using LR. Extending to multiclass classification (e.g. child_yes, adolescent_no) on the merged dataset, SVM attained a peak accuracy of 99.55%, while LR achieved 95.90% accuracy, RF achieved 94.10%, and NB achieved 82.50%. Further enhancement by grid search hyperparameter tuning yielded SVM with 99.55% accuracy, LR with 99.09%, and RF with 96.82%. The study concludes that multiclass classification models tailored to age subgroups substantially improve diagnostic accuracy and could serve as a foundation for scalable ASD screening tools.

9) Kunda et al. [18] proposed an ML model to classify autism using the ABIDE-I dataset, which comprises a total of 1,035 cases (505 ASD, 530 TC) of resting-state fMRI and phenotypic data. The study proposed the tangent Pearson embedding method for extracting second-order features. It also implemented an approach for domain adaptation to enhance classification accuracy. The model was trained using Ridge, LR, and SVM, and its performance was evaluated through ten-fold and leave-one-out cross-validation. Ultimately, it achieved an accuracy of 73% with the Ridge classifier.

The reviewed studies collectively highlight significant advancements in AI-driven ASD detection and diagnosis. They show the potential of AI to significantly enhance and facilitate the early detection and diagnosis of ASD through the application of ML algorithms and dimensional reduction techniques. Table I summarizes the discussed studies for ASD detection from an AI perspective, and the symbol "*" indicates ML algorithm(s) that achieved the highest accuracy.

TABLE I. COMPARATIVE TABLE FOR ASD STUDIES USING PHENOTYPIC DATA

Study	Dataset size	ML algorithms	The highest accuracy
Perochon et al. [9]	475	XGBoost*	87.8%
Mazumdar et al. [10]	400	TreeBagger*	59%
Alkahtani et al. [12]	2,940	SVM, LR, RF, KNN, DT, MLP, GBoost, CNN*	92%
Raj et al. [14]	1,100	SVM, LR, NB, KNN, NN, CNN*	99.53%
Akter et al. [15]	2,009	AB, FDA, DT, Glmboost, LDA, MDA, PDA, CART, SVM*	98.77%
Erkan et al. [16]	1,100	KNN, SVM, RF*	100%
Bawa et al. [17]	1,100	LR, DT, RF, NB, KNN, SVM*	99.55%
Kunda et al. [18]	1,035	LR, SVM, Ridge*	73%

Although numerous datasets, studies, and approaches, ML approaches demonstrated high potential for identifying ASD from behavioral, physiological, and neurocognitive data. There

are several common limitations:

- Data limitations:
  - Small sample sizes: Many studies were conducted on limited-scale datasets, particularly those focusing on specific subpopulations (e.g. children, females, or multi-site studies). This scarcity of data reduces the statistical power and representativeness of models, undermining the reliability of their results and classification accuracy.
  - Dataset imbalance: ASD datasets are often highly imbalanced, leading to bias in ML models.
  - Variability across sites: Multi-site neuroimaging studies (e.g. fMRI) showed high inter-site variability due to differences in scanner protocols, demographic diversity, and sample heterogeneity.
  - Lack of standardized datasets: Many ML models were trained on custom datasets, making it challenging to compare results across studies.
  - Lack of detailed phenotypic data, especially regarding ASD severity and cognitive variability.
  - A lack of diverse participants (e.g. in age, gender, or ethnicity) further means models may be overfit to narrow populations.
  - Need for real-world testing: real-world data variability is untested.
- Model generalization:
  - Overfitting on small datasets: ML models tend to overfit when trained on limited or highly structured datasets.
  - Poor generalization to real-world clinical settings: Many ML models were evaluated only on academic datasets without validation in clinical environments.
  - Feature selection dependence: Some studies heavily relied on specific feature selection methods, which might not generalize well to different datasets.
- Computational challenges:
  - High computational cost: Some ML models require significant computing resources, making them infeasible for real-time clinical applications without further optimization.
  - Interpretability of AI models: Most ML models for ASD detection operate as "black boxes," offering little explanation for their decisions. This opacity makes it difficult for clinicians to trust and implement them in real-world diagnostic settings.

III. Intelligent Mental Healthcare Model

The model consists of two main pipelines, as shown in Fig. 1. The model initiates by collecting the datasets and then processing them through the data pipeline using ML methods. Subsequently, the model inputs the resulting dataset into the learning pipeline, in which a supervised ML algorithm is applied to develop a trained computational mental model for ASD detection. The following sections focus on the details of the data and the learning pipelines, respectively.

A. Data Pipeline

The data pipeline includes four primary sequential steps: data collection, integration, inspection, and preprocessing, as shown in Fig. 1. First, a unified dataset was created by merging data sources collected from many international sites that dealt with autism and employed various examinations and procedures for making diagnoses. Afterwards, data preprocessing techniques were employed on the dataset to enhance its quality and format after a thorough inspection. Furthermore, examine whether the dataset is balanced.

1) Data collection: The model utilized the phenotypic data in ABIDE-II [19] collected from over nineteen international sites, as it provides more detailed phenotypic data than any other dataset. As defined, the phenotypic data in ABIDE-II present 1,114 real-world cases aged 5 to 64 years. The dataset contains 521 ASD cases and 593 TC cases. Each case is represented by 347 structured characteristics and clinical data, such as IQ, medical history-related questions, demographic variables, diagnostic status, and an extensive array of behavioral assessments. Additionally, patients' basic information, including gender, age, handedness index score, and handedness category, is utilized.

2) Data integration and inspection: Upon downloading the phenotypic data from all nineteen sites participating in the ABIDE-II dataset, they were comprehensively analyzed. The individual datasets were concatenated into a unified file to ensure consistency and uniformity across the various sites. And then, inspecting the data to find structural issues, leading us to the next stage of applying data preprocessing techniques.

3) Data preprocessing: Crucial preprocessing techniques were applied to clinch the subsequent analyses quality and reliability and maintain and improve data balance.

a) Data cleaning is essential to ensure the data quality. Data cleaning operations need to be done very carefully and precisely so as not to end up with too little data or too much data with useless values. These operations were carried out in several steps as illustrated in the experimental results section.

b) Data balancing is crucial when the number of instances in different classes is significantly skewed as this will lead to biased model performance; where the model may be biased towards the majority class and the minority class may be significantly underrepresented making it challenging for the AI model to learn patterns effectively, resulting in poor performance on the minority class. In the ABIDE-II dataset, after applying data cleaning and removing some observations, the ground truth in all remaining data was checked. The ratio between ASD and TC observations is 1:1.08, which means they are nearly equal in size and the data is balanced.

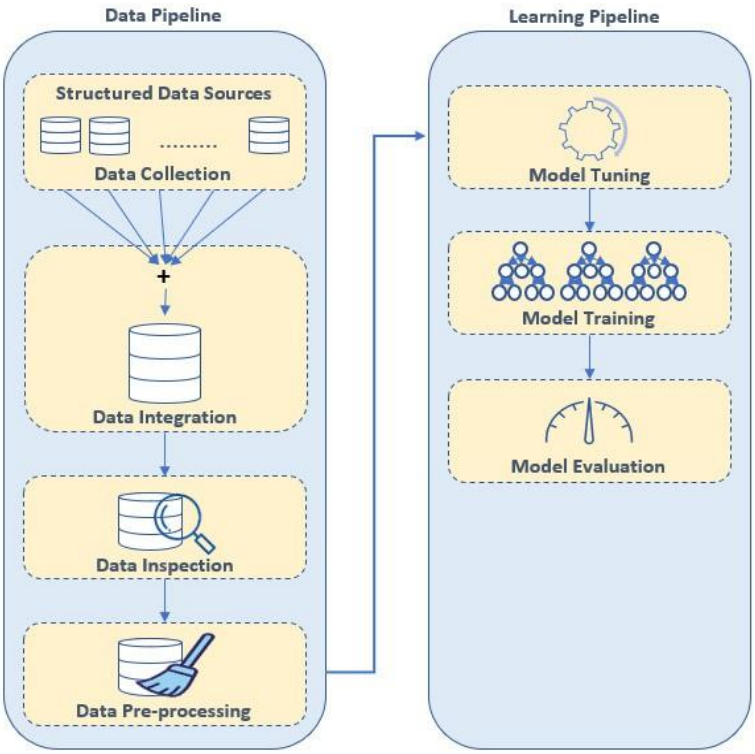


Fig. 1. The intelligent mental healthcare model for ASD detection.

B. Learning Pipeline

Model tuning, model training, and model evaluation are the three successive stages that comprise the learning pipeline to develop a trained model for detecting ASD, as shown in Fig. 1. The model tuning is an iterative process that plays a substantial role in identifying the optimal values of the essential parameters associated with the selected supervised learning algorithm. Each iteration involves setting the other parameters to a constant value and testing the targeted parameter over a predetermined range of values. This process continues until all parameters have reached the optimal values. After that, the model is trained and evaluated using the supervised learning algorithm based on the optimal values of its parameters.

1) Model tuning: For the model, the RF algorithm was selected for its promising results in various healthcare studies with high-dimensional data. RF is an ensemble learning method that combines the predictions of multiple decision trees and aggregates their predictions, providing robustness and enhancing accuracy in the required task [5]. To achieve optimal performance, it is essential to tune the values of five distinct parameters:

- Two of them are for the preprocessing and dimensional reduction, which are:
  - The percentage of missing cells in a column at which this column will be deleted if it exceeds this percentage.
  - The percentage of testing data and training data to divide the dataset.
- The other three parameters that need to be tuned for setting the RF algorithm are:
  - In-bag-fraction, which determines the fraction of observations that are randomly selected with replacement for each bagged tree.
  - The number of trees in the forest.
  - The number of predictors to sample, which is the number of feature variables used to pick at random for each decision split at each node.

Five rounds were conducted to tune these parameters and select their optimal values, as shown in Fig. 2. A round was conducted to decide the best value for each parameter. In each round, the targeted parameter is tested with a given range of values while the other four parameters are given a fixed value. At the end of each round, the value that scored the highest accuracy is used to set this parameter in the following rounds, and so on over all the rounds till the best-scoring value for each parameter is determined.

2) Model training and evaluation: The model input training data comprises a feature matrix that includes instances representative of individuals with ASD as well as instances of typical control subjects; each instance is characterized by meticulously selected features. Subsequently, a testing phase is conducted utilizing a dataset that has not been previously encountered by the model, during which the model's performance is assessed through appropriate evaluation metrics. Ultimately, when presented with any new instance, the model is expected to identify individuals with ASD based on their respective features.

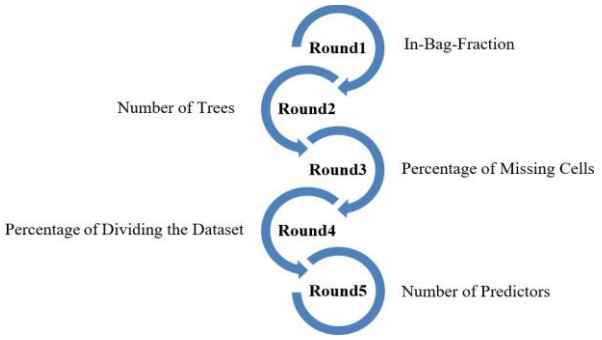


Fig. 2. Five sequential rounds of model tuning.

IV. Experimental Results

A. Dataset

The proposed model utilized ABIDE [5], an extensive electronic database of brain scans of individuals with ASD. More than 20 different sites shared in collecting this database to create two large-scale collections, ABIDE-I [20] and ABIDE-II [19]. Both collections provide resting state functional MRI (R-fMRI) along with phenotypic data for individuals with ASD and individuals with typical controls. The ABIDE-I [20] collection was established across 17 international brain imaging sites. ABIDE-I contains 1,112 cases, ranging between 539 ASD cases and 573 typical control cases.

The second collection, ABIDE-II [19], was publicly released in June 2016. Nineteen different international sites collaborated to donate 1,114 cases, ranging between 521 ASD cases and 593 typical control cases. After using ABIDE-I, many challenges and obstacles were found in it, such as the complex structure of the brain connectivity and the major heterogeneity of ASD; those issues were mapped for the essential need of a larger dataset that contains samples with more distinct characteristics [5], [21]. ABIDE-II was released to resolve these challenges and includes samples with better characteristics, features, and psychiatric variables [19]. Table II presents all 19 sites with their size and configuration.

1) Phenotypic data: The phenotypic data, in the ABIDE-II release, include a wide array of demographic, diagnostic, and behavioral measures for each participant. Many features are standardized scores from validated instruments for comparison across ages or test versions. In addition, patients' demographics and basic information, such as age, sex, handedness category, and handedness index score, were used in ABIDE-II. Around sixteen assessments were used in diagnosing ASD. They are as follows, highlighting the most essential features and their type and range that were used to build the dataset:

- Cognitive Ability (Intelligence Quotient – IQ): Verbal IQ, Performance IQ, and Composite IQ tests are administered and result in continuous scores in the range of 45-156.
- Assessments of social interaction, communication, restricted and repetitive behaviors, abnormal early development, stereotyped behaviors and restricted interests, imagination/creativity, social affect, and executive functions:
  - Autism Diagnostic Interview–Revised (ADI- R)
  - Autism Diagnostic Observation Schedule (ADOS)
  - Social Responsiveness Scale (SRS)
  - Social Communication Questionnaire (SCQ) and Autism Quotient
  - Repetitive Behavior Scale–Revised (RBS-R)
  - Behavior Rating Inventory of Executive Function (BRIEF)
  - Child Behavior Checklist (CBCL)
  - Adaptive Behavior (Vineland-II).

All scores resulting from these assessments are continuous numerical values. Higher scores indicate more ASD-related symptoms.

- Anxiety (Multidimensional Anxiety Scale for Children – MASC): MASC is a self-report questionnaire (child or adolescent) assessing anxiety symptoms. All scores are T-scores (mean=50, SD=10) for the MASC sub-scales: Physical Symptoms, Harm Avoidance, Social Anxiety, and Panic, plus an overall total and an Anxiety Disorder Index.
- Depression (Beck Depression Inventory – BDI): BDI examines depression symptoms and gives a score in the range (0-63) for each symptom.
- Language ability (Clinical Evaluation of Language Fundamentals – CELF): The CELF is a standardized language ability test. ABIDE II includes standard scores for Core Language, Receptive, and Expressive language abilities. Higher scores indicate better language skills.
- Other medical/behavioral details: Medication status indicates if the participant was on any psychotropic medications around the time of the scan and what they were (commonly ADHD meds, SSRIs, etc.). Body Mass Index (BMI) is included as an essential health metric.

B. Data Integration, Inspection, and Preprocessing

The ABIDE-II individual datasets were concatenated into a unified file to ensure consistency and uniformity across the various sites. After the investigation, the first issue was the large amount of missing data. The data contains 347 features; the first two columns are unique identifiers. The third is the ground truth or the label for each entry that is filled with value 1 for ASD or 2 for TC (which changed after that to 0, Fig. 3 shows the difference before and after changing labels), and by which the dataset is divided into two classes ASD with 521 records and TC with 593 records. The remaining 344 columns are different features that are responses from the patient to the doctor's questions. By inspecting the data, many structural issues were found, leading us to the next stage of applying other data preprocessing techniques.

TABLE II. PHENOTYPIC DATA ON ABIDE-II FROM DIFFERENT SITES

Site name	Samples Total size	Size of ASD	ASD Age Range	Size of TC	TC Age Range
Barrow Neurological Institute	58	29	18 - 62	29	18 - 64
Erasmus University Medical Centre	54	27	6 - 11	27	6 - 10
Neural Control of Movement Lab	37	13	15 - 27	24	14 - 31
Georgetown University	106	51	8-13	55	8-13
Indiana University	40	20	17-54	20	19-37
Institut Pasteur and Robert Debre Hospital	56	22	6-27	34	8-47
Katholieke Universiteit Leuven	28	28	18-25	-	-
Kennedy Krieger Institute	211	56	8-13	155	8-13
NYU Langone Medical Center Sample 1	78	48	5-34	30	5-23
NYU Langone Medical Center Sample 2	27	27	5-8	-	-
Olin Neuropsychiatry Research Center	59	24	18-31	35	18-31
Oregon Health and Science University	93	37	7-15	56	8-14
Trinity Centre for Health Sciences	42	21	10-20	21	15-20
San Diego Stat University	58	33	7.4-18	25	8.1-17.7
Stanford University	42	21	8-13	21	8-13
University of California Davis	32	18	12-18	14	12-17
University of California Los Angeles	32	16	8-15	16	8-14
University of Miami	28	13	7-13	15	7-13
University of Utah School of Medicine	33	17	9-39	16	11-16
Total	1,114	521	5-62	593	5-64

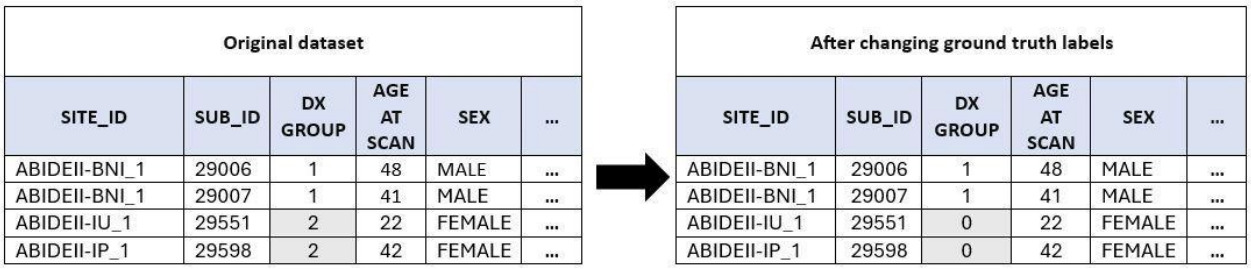


Fig. 3. Sample for the dataset after changing ground truth labels.

1) Removing duplicate observations. By observing the data, no duplicate records were found.

2) Handling missing data. Either by filling with appropriate values or removing the entire observation; the choice depends on the nature of the missing data and the specific requirements of the model.

3) Filter undesired outliers. Some data ranged from -100 to 100; we manipulated these values by shifting them by 100 to acquire non-negative values ranging from 0 to 200.

4) Manipulating nonnumeric values. The raw data contains six columns with a string data type. All string data has to be converted into numeric values to fit the data into the model. The Levenshtein distance algorithm was utilized for this task. In each column, the largest string value was picked and used as the reference string in the algorithm, and each string was replaced with the distance evaluated by the Levenshtein algorithm. Using the MATLAB tool, a function named convert_txt2num was written that takes a vector, which is a string column from the dataset, and it finds the longest string, then calls the implemented Levenshtein algorithm, which takes the string vector and the longest string as the reference value for the algorithm. The Levenshtein function returns a numeric vector that is used to replace the string vector. Algorithm 1 shows the pseudocode for the implemented convert_txt2num function that leverages the Levenshtein algorithm.

Algorithm 1: Function convert_txt2num that uses the Levenshtein algorithm

```
lenVec ← length of each string in the vector (v)
maxInd ← max(lenVec)
        longestStr ← v(maxInd) for each s in v do
numericVector(index(s)) = Levenshtein(s, longestStr)
end for
```

C. Model Tuning

After performing all the steps of the data pipeline, as mentioned previously, five different parameters have to be tuned, two of them for data cleaning and preprocessing steps,

which are the percentage of missing cells in a column at which this column will be deleted if it exceeds this percentage, and the percentage of testing data for training data to divide our dataset. Regarding the other three parameters need to be tuned are for the building the RF model which are: number of used trees in the forest and the in bag fraction which is determining the fraction of observations that are randomly selected with replacement for each bagged tree and the last parameter is the number of predictors to sample which is the number of feature variables used to pick at random for each decision split at each node. To tune these parameters and select their best values, we created five rounds: a round to decide the best value for each parameter. In each round, the targeted parameter is tested with a given range of values while the other four parameters are given a fixed value. At the end of each round, the value that scored best is used to set this parameter in the following rounds, and so on over all the rounds till the best-scoring value for each parameter is determined.

5) Round 1: The first parameter to tune is the In-Bag-Fraction parameter for the RF model. This parameter refers to the number of instances that will be used in each bag/tree in the forest to train with them (sample size drawn randomly for the training of each bag/tree). The value is a fraction of the whole number of available instances in the training data. Setting this parameter with a fraction less than one means that each tree will train with a subset of the training data that is selected randomly. So, decreasing the covariance and avoiding the over-fitting problem. Fig. 4 shows the effect of changing this fraction on the model accuracy. The other parameters were set to their default values, which are 64 trees, 20% of the data for testing, and 80% for training. If the column contains more than 50% empty cells, then delete this column/feature. And finally, the number of predictors to choose from each node was set by the square root of the number of features, which is the default value given by Breiman [22]. By observing the accuracy in Fig. 4, the highest value for the accuracy was achieved by using the fraction 0.6. So, it is the optimal value to set this parameter in the model.

6) Round 2: The second parameter to tune is the number of trees used to build the forest. A study was published to determine how many trees should be found in the RF, and it stated that the best values will be found in the range (64-128) [23]. Training multiple trees is one of the most essential features of RF; a few trees can result in bad predictions, and a vast number can cause performance issues. To tune this parameter, the value of all other parameters was set as follows: 20% of the dataset for testing, the column is deleted if it contains 50% or more empty cells, and the number of predictors to choose from on each node is the square root of the number of features. Finally, the in-bag fraction was set with the best value from round 1, which is 0.6. The model was tested with a broader range, which is for a minimum of 30 trees and a maximum of 200 trees. By observing the rate of change of the model accuracy by changing the number of trees in the RF in Fig. 5, the optimal value to tune the number of trees parameter is 100 trees.

7) Round 3: The third parameter to tune is a dataset-related parameter, which is the percentage of empty cells that can be found in a column. This round aims to determine the acceptable percentage limit for empty cells within each feature, without losing significant information in the dataset. After the first two rounds, the in-bag fraction and the number of trees were tested for their best values, which were 0.6 and 100, respectively. In this round, the data will be split into testing and training sets in a 20:80 ratio, with the number of predictors set to the square root of the total predictors identified in the input data. The required parameter was tested using percentages ranging from 100% to 10%. Fig. 6 shows the accuracy assessed at each percentage value. As depicted in Fig. 6, the peak accuracy occurs at 70%, indicating that any column with over 70% of its cells empty will be removed, consequently decreasing the number of feature vectors in the dataset.

8) Round 4: The fourth parameter to tune is the percentage needed from the dataset to be used for testing and the rest for training. Determining the optimal percentage for this division directly impacts the performance of the model, as it needs a suitable amount of data that yields robust results and a scalable model. Ensuring enough data for training and testing can prevent problems such as overfitting -if too much data is used for training- and underfitting -if too little data is used for training- and so balance is achieved between model generalizability and its predictive power. The round started with a ratio of 15:85, then increased it to 20:80, then 25:75, and finally 30:70. Fig. 7 shows the variation in accuracy through these different ratios. The other parameters were set by the following values: 64 trees, in-bag-fraction was 0.6, and the number of predictors selected randomly at each node was 71. If the column contains more than 70% empty cells, then delete this column/feature. The experiment showed that allocating 20% of the data for testing and 80% for training gives the model the highest accuracy, so the ratio 20:80 was chosen to tune this parameter.

9) Round 5: The fifth parameter to determine is the number of predictors selected randomly at each node during tree construction to choose the best split from them, denoted by variable m. It is a critical parameter as it controls the randomness of feature selection and influences the diversity of the model, ensuring robustness and reliability. This parameter should be carefully adjusted due to its effect on the model performance, as high values of m might cause high computation cost, but low values might also reduce model diversity or even cause overfitting. The model was tested with m values ranging from 2 to 20, and its accuracy was observed. Fig. 8 clearly determined that m=5 scored best, so it will be used in the final model training phase.

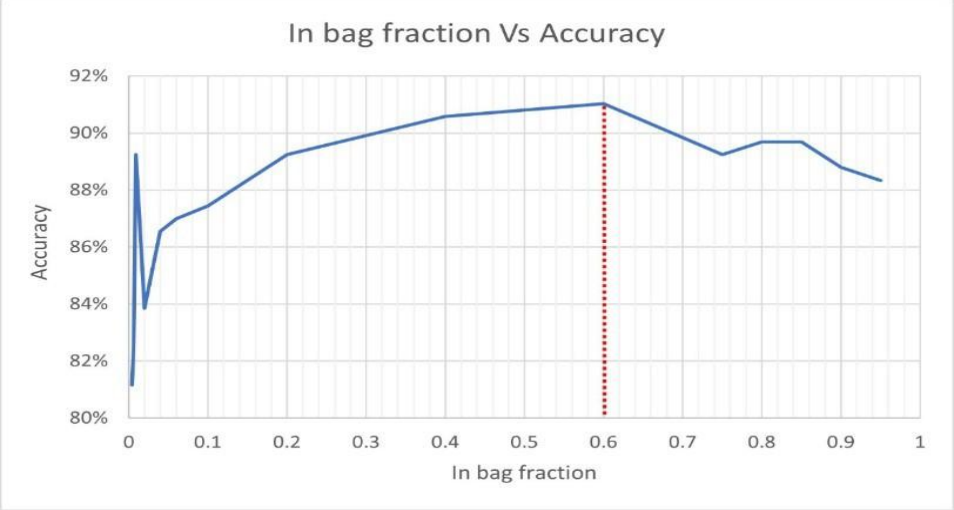


Fig. 4. Tuning the InBagFraction parameter for the random forest.

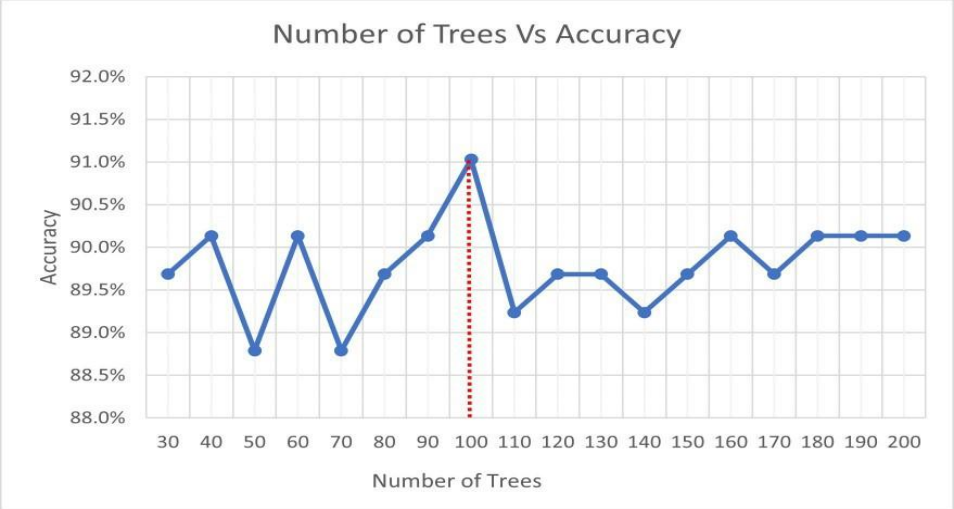


Fig. 5. Tuning the number of trees required to build the forest.

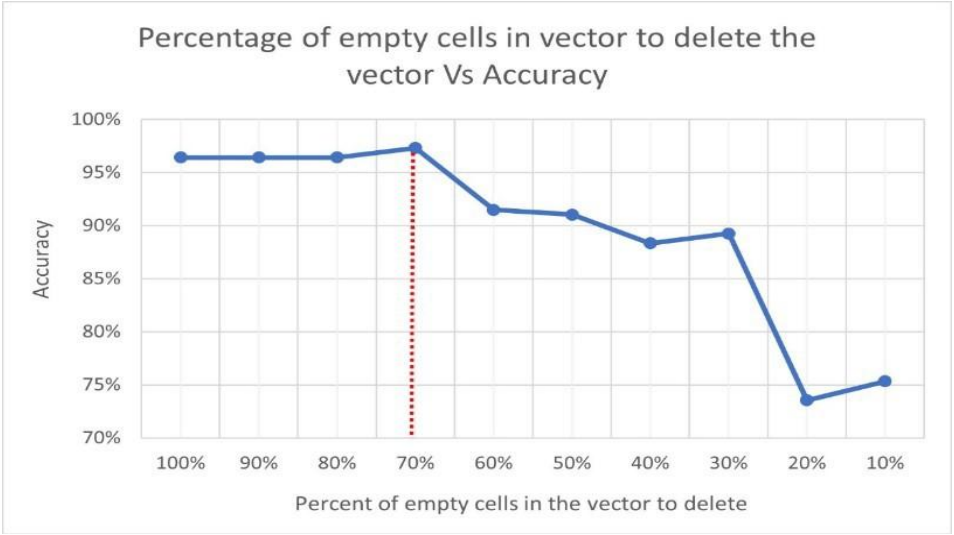


Fig. 6. Tuning the percentage of empty cells.

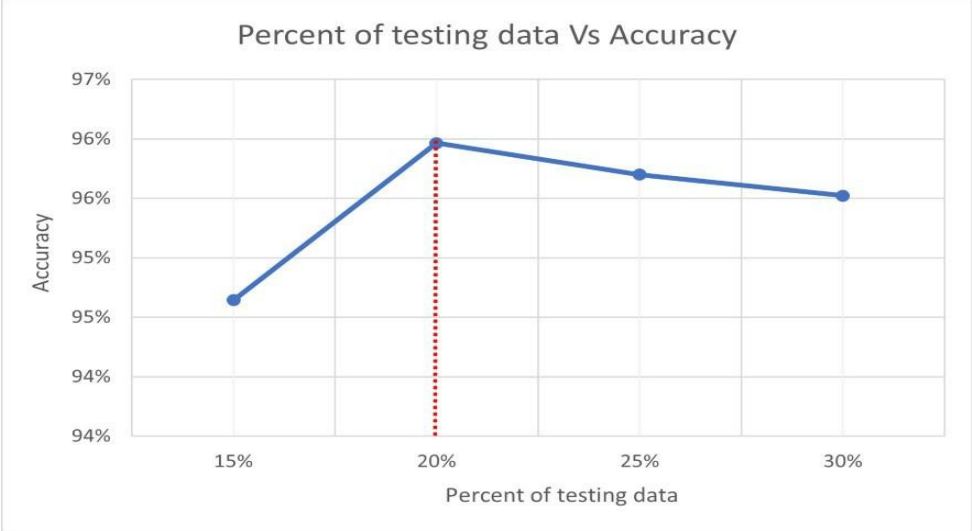


Fig. 7. Tuning the percentage of testing data from the original dataset.

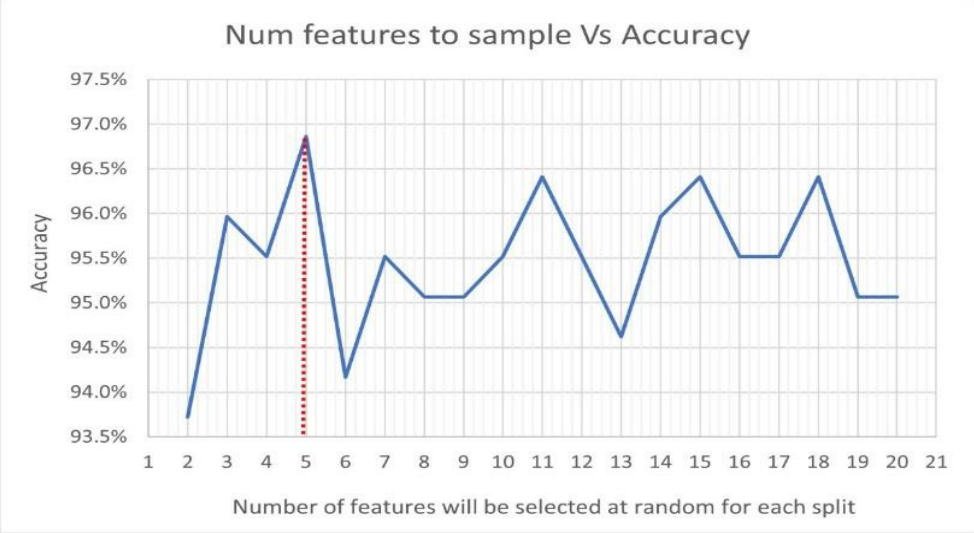


Fig. 8. Tuning the number of predictors selected randomly at each node.

V. Results and Discussion

This research proposed an intelligent model with a tuning phase based on the ML pipeline and systematically optimized five key RF parameters: the number of trees, in-bag-fraction, percentage of training and testing data, number of features selected randomly at each node, and percentage of empty cells in which a column will be deleted. This optimization phase proved critical, as it significantly enhanced the classifier's performance compared to default configurations. The experimental results demonstrated that the optimized RF classifier performed with high accuracy after applying the tuning phase. Furthermore, to assess the RF model against other classifiers' accuracies, multiple ML classifiers were employed to evaluate the classification performance on the ABIDE-II phenotypic dataset. The models used include Random Forest RF, Support Vector Machine, K-Nearest Neighbors, Decision Tree, Fine Gaussian SVM, Weighted KNN, and Random Forest integrated with Principal Component Analysis. MATLAB served as the primary programming platform for data handling, model development, and performance evaluation. The configurations for each model are detailed below:

1) Random forest: The Random Forest classifier was constructed using thirty decision trees. At each decision node, a random subset of predictors was selected, with the number of predictors set to the square root of the number of observations. The maximum number of splits per tree was set equal to the total number of observations, allowing for fully grown trees. These parameters were selected to balance model complexity with computational efficiency while maintaining robust performance across the training and testing sets.

2) Support vector machine: A linear kernel function was employed to construct the separating hyperplane. This configuration was chosen for its simplicity and efficiency in high-dimensional spaces.

3) Fine K-Nearest neighbors: This model used a single nearest neighbor (k=1) with the Euclidean distance metric. The distance weight was set to automatic, allowing the algorithm to select the most suitable weighting scheme based on the dataset.

4) Decision tree: The decision tree classifier was limited to a maximum of 100 splits. Gini's diversity index was used as the split criterion to evaluate the quality of each node partition.

5) Fine Gaussian support vector machine: A Gaussian (radial basis function) kernel was utilized to allow for nonlinear decision boundaries in the feature space.

6) Weighted K-Nearest neighbors: This variant of the KNN algorithm used ten nearest neighbors (k=10) with the Euclidean distance metric. Distance weights were computed using a squared inverse function, giving greater influence to nearer neighbors during classification.

7) Random forest with principal component analysis: Dimensionality reduction was performed using PCA prior to training the RF model. PCA components were selected to retain 95% of the total variance in the dataset. The RF classifier was configured identically to the standard RF model, with thirty trees, the square root of the number of observations as the number of predictors sampled at each node, and a maximum of splits equal to the number of observations.

The models attained accuracies of 95.06% with RF, 93.27% with SVM, 91.29% with KNN, 94.61% with DT, 77.47% with fine Gaussian SVM, 91.11% with weighted KNN, and 84.02% with RF+PCA, as shown in Fig. 9. The optimized RF model achieved the highest classification accuracy of 96.86% across all models. It consistently outperformed other classifiers. This finding highlights the importance of the model tuning stage and aligns with and reinforces observations from the literature, which often report superior performance of ensemble methods in similar high-dimensional healthcare datasets.

Moreover, this research underscored the pivotal role of data preprocessing, particularly data cleaning methods and dimensionality reduction, in improving model accuracy and generalization. This finding is consistent with previous studies that have highlighted how poor data preparation can severely degrade model performance, regardless of the classifier used.

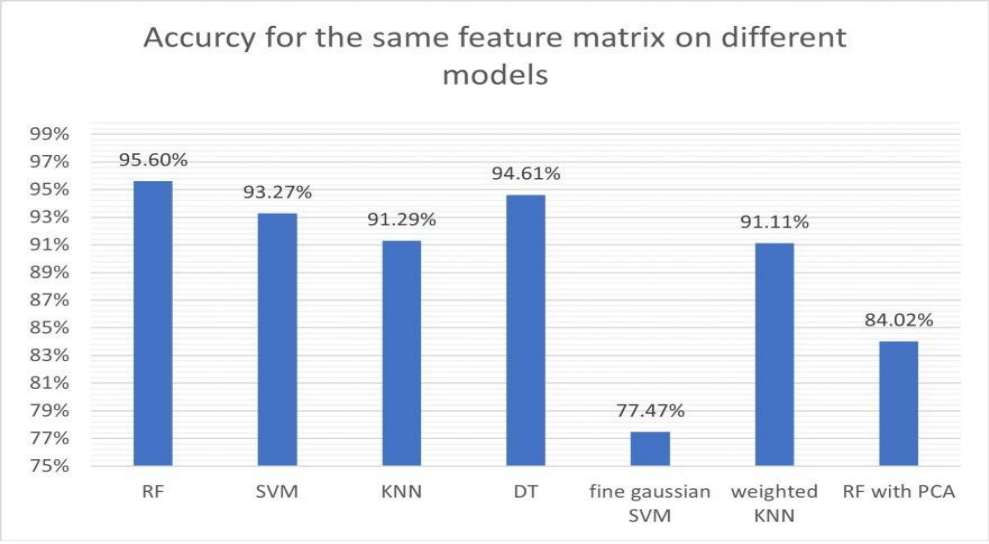


Fig. 9. Accuracy of different ML models on the same feature matrix.

VI. Model Validation and Empirical Confirmation

To further validate the predictive capabilities of the model and confirm its real-world applicability, an empirical validation study was conducted. While the model demonstrated high accuracy metrics on the ABIDE-II dataset, a separate qualitative approach was employed to assess its performance on a set of independent, non-structured data points. This methodology was designed to simulate a real-world diagnostic scenario, moving beyond the quantitative features of the training data.

A search for anecdotal accounts and personal narratives from parents describing their children's autistic traits was performed. These qualitative descriptions, sourced from publicly available forums and support group websites, provided a rich, narrative-based perspective on the diagnostic process. Five distinct case studies were curated, each representing a unique constellation of behaviors and characteristics commonly associated with ASD.

The narratives were meticulously analyzed to extract values for the key features utilized by the model. This process involved a careful interpretation of the qualitative data, mapping descriptive language to the quantitative feature scales used during model training. For example, a parent's description of a child's intense focus on specific objects and resistance to change was translated into a high value for the "restricted and repetitive behaviors" feature.

The conversion of unstructured, qualitative narrative data into the quantitative, structured feature vector required by the trained RF model was accomplished using Gemini [24], a Large Language Model (LLM), as an advanced research agent

for automated feature extraction. The process involved a highly specific Prompt Engineering strategy:

- Input: The de-identified parental narrative was supplied as the primary unstructured text.
- Instruction set: The prompt explicitly instructed Gemini to act as a clinical coder and to map the descriptive language onto the precise quantitative features used in the ABIDE-II training data.
- Scale definition: The prompt provided the LLM with a defined, ordinal scale (e.g. 1-5 or a binary 0/1) for each feature, compelling the model to translate subjective descriptions (e.g. "rarely makes eye contact") into a numerical rating (e.g. "Social Reciprocity =4"). After that, the extracted numerical values were manually validated to avoid the potential for LLM Hallucinations or the ambiguous nature of subjective human language.
- Output format: The LLM was instructed to output the extracted feature values in a structured format (e.g. a delimited list) to facilitate direct integration into the classification model.

This utilization of an LLM allowed for the rapid and systematic transformation of complex qualitative data into a format suitable for computational analysis, a process that traditionally requires extensive manual human coding.

The extracted feature values for each of the five cases were then input into the trained model. The model's classification output for each case was subsequently compared against the known diagnosis (ASD vs TC) from the original narrative. The model correctly classified four out of the five cases, yielding an accuracy rate of 80% on this small, independent dataset.

This successful classification rate on qualitative data provides a robust, real-world confirmation of the model's accuracy and effectiveness. It demonstrates the model's ability to generalize beyond the structured data of the ABIDE-II database and accurately interpret complex, narrative-based descriptions of ASD symptoms. While this validation is not a substitute for a large-scale clinical trial, it provides strong evidence of the model's potential utility in a preliminary screening context.

VII. Conclusion

This research aimed to advance the field of ASD detection by proposing a data-driven, ML-based framework, with a specific focus on the optimization of the RF classifier. It contributes to the growing body of work exploring the role of AI in the early and accurate identification of ASD, a complex neurodevelopmental condition characterized by a wide spectrum of symptoms. The proposed model highlighted the potential of leveraging a comprehensive set of phenotypic features and applying an ML algorithm to develop a reliable and accurate model, which can guide researchers in their studies. Healthcare professionals can also use it to assist them in making informed decisions and improving the efficiency of ASD screening processes.

Phenotypic data, encompassing a wide range of observable traits and characteristics, serve as valuable indicators for understanding the heterogeneity of ASD. Behavioral, cognitive, and physiological features have been extensively studied to identify patterns that differentiate individuals with ASD from their typically developing counterparts. These phenotypic markers often include social communication abilities, repetitive behaviors, sensory sensitivities, and developmental milestones. Research in the field has emphasized the importance of a comprehensive and multidimensional approach to phenotypic characterization, considering the diverse manifestations of ASD across individuals.

This research systematically adjusted the key hyperparameters for RF optimization. The experimental results spot the optimal values to tune the model and its input data which were: 0.6 for in-bag-fraction, 100 trees to build the forest, delete columns with more than 70% empty cells, number of features selected at random for each split is 5 and finally divide the dataset with 20% for testing and 80% for training. By tuning the model with those values, it achieved promising results with a classification accuracy of 96.86%, a sensitivity of 97.14%, and a false positive rate of 3.39%. Achieving 96.86% accuracy on the ABIDE-II dataset suggests that there are detectable signatures of ASD in phenotypic data that ML can pick up. Furthermore, this study provides evidence that ML can be effectively applied to large multi-site phenotypic datasets to identify individuals with autism based on behavior and medical patterns. This model presents a scalable and expedited alternative to current models, thereby aligning effectively with the clinical demand for efficient diagnostic tools.

The empirical validation study successfully bridges the gap between the RF model's high quantitative performance on structured datasets and its real-world applicability. By successfully translating qualitative, narrative-based descriptions into actionable data, the study demonstrated the model's ability to generalize beyond its training data and accurately interpret complex, non-structured information. The resulting 80% accuracy on this small, independent dataset provides strong evidence of the model's potential utility as a preliminary screening tool. While not a substitute for comprehensive clinical evaluation, this validation confirms the model's robustness and its capacity to assist in the early identification of ASD in practical, diagnostic scenarios, thereby paving the way for future clinical integration and large-scale validation studies.

While this study demonstrates the efficacy of the proposed approach, it is important to acknowledge certain limitations. Firstly, the performance of the model heavily relies on the quality and availability of phenotypic data. Obtaining comprehensive and standardized data across different populations and settings may pose challenges. Additionally, the generalizability of the model to diverse populations needs to be explored further to ensure its applicability in different cultural and geographical contexts.

Several avenues for future research warrant consideration. The integration of additional data sources and the incorporation of multimodal data, such as genetic and neuroimaging data, to further enhance the accuracy and predictive power of the

model. Explore hybrid AI models that combine ML models (e.g., SVM + RF) with DL architectures (e.g. CNN + Transformers) for improved accuracy. Moreover, the development of user-friendly tools and applications based on the model's findings could facilitate widespread adoption and implementation in clinical settings. Continued research and innovation in this field are crucial to further advance our understanding and detection capabilities for ASD.

DATA AVAILABILITY

The ABIDE dataset used in our current study is publicly available on the Autism Brain Imaging Data Exchange repository. https://fcon_1000.projects.nitrc.org/indi/abide/fornon-commercial research purposes.

REFERENCES

[1] "World health organization." https://www.who.int/news-room/fact-sheets/detail/autism-spectrum-disorders. (accessed Dec. 28, 2024).

[2] R. P. Walensky, R. Bunnell, J. Layden, C. K. Kent, A. J. Gottardy, M. A. Leahy, et al., "Prevalence and Characteristics of Autism Spectrum Disorder Among Children Aged 8 Years-Autism and Developmental Disabilities Monitoring Network, 11 Sites, United States, 2018," tech. rep., United States, 2021.

[3] R. G. Schwartz, Handbook of child language disorders: Second edition. Exon Publications, 2017.

[4] J. F. Underwood, K. M. Kendall, J. Berrett, C. Lewis, R. Anney, M. B. Van Den Bree, et al., "Autism spectrum disorder diagnosis in adults: Phenotype and genotype findings from a clinically derived cohort," *British Journal of Psychiatry*, vol. 215, no. 5, pp. 647–653, 2019.

[5] "Autism Brain Imaging Data Exchange." https://fcon 1000.projects.nitrc.org/indi/abide/. (accessed Dec. 28, 2024).

[6] A. Di Martino, D. O'connor, B. Chen, K. Alaerts, J. S. Anderson, M. Assaf, et al., "Enhancing studies of the connectome in autism using the autism brain imaging data exchange ii," *Scientific data*, vol. 4, no. 1, pp. 1–15, 2017.

[7] N. S. El-Askary, M. A. Salem, and M. I. Roushdy, "Feature extraction and analysis for lung nodule classification using random forest," *ACM International Conference Proceeding Series*, pp. 248–252, 2019.

[8] N. S. El-Askary, M. Mabrouk Morsey, A. M. Mahmoud, and T. I. El-Arif, "On Development of Computer Aided System for Detecting Brain Neurological Disorders," *Proceedings - 11th IEEE International Conference on Intelligent Computing and Information Systems, ICICIS 2023*, pp. 94–101, 2023.

[9] S. Perochon, J. M. Di Martino, K. L. Carpenter, S. Compton, N. Davis, B. Eichner, et al., "Early detection of autism using digital behavioral phenotyping," *Nature Medicine*, vol. 29, no. 10, pp. 2489–2497, 2023.

[10] P. Mazumdar, G. Arru, and F. Battisti, "Early detection of children with Autism Spectrum Disorder based on visual exploration of images," *Signal Processing: Image Communication*, vol. 94, no. January, p. 116184, 2021.

[11] J. Gutie´rrez, Z. Che, G. Zhai, and P. Le Callet, "Saliency4asd: Challenge, dataset and tools for visual attention modeling for autism spectrum disorder," *Signal Processing: Image Communication*, vol. 92, p. 116092, 2021.

[12] H. Alkahtani, T. H. Aldhyani, and M. Y. Alzahrani, "Deep learning algorithms to identify autism spectrum disorder in children-based facial landmarks," *Applied Sciences*, vol. 13, no. 8, p. 4855, 2023.

[13] "Kaggle." https://www.kaggle.com/. (accessed Dec. 28, 2024).

[14] S. Raj and S. Masood, "Analysis and detection of autism spectrum disorder using machine learning techniques," *Procedia Computer Science*, vol. 167, pp. 994–1004, 2020.

[15] T. Akter, M. S. Satu, M. I. Khan, M. H. Ali, S. Uddin, P. Lio, et al., "Machine learning-based models for early stage detection of autism spectrum disorders," *IEEE Access*, vol. 7, pp. 166509–166527, 2019.

[16] U. Erkan and D. N. Thanh, "Autism Spectrum Disorder Detection with Machine Learning Methods," *Current Psychiatry Research and Reviews*, vol. 15, no. 4, pp. 297–308, 2019.

[17] P. Bawa, V. Kadyan, A. Mantri, and H. Vardhan, "Investigating multiclass autism spectrum disorder classification using machine learning techniques," *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, vol. 8, no. May, p. 100602, 2024.

[18] M. Kunda, S. Zhou, G. Gong, and H. Lu, "Improving Multi-Site Autism Classification via Site-Dependence Minimization and Second-Order Functional Connectivity," *IEEE Transactions on Medical Imaging*, vol. 42, no. 1, pp. 55–65, 2023.

[19] A. Di Martino, D. O'Connor, B. Chen, K. Alaerts, J. S. Anderson, M. Assaf, et al., "Enhancing studies of the connectome in autism using the autism brain imaging data exchange II," *Scientific Data*, vol. 4, pp. 1–15, 2017.

[20] A. Di Martino, C. G. Yan, Q. Li, E. Denio, F. X. Castellanos, K. Alaerts, et al., "The autism brain imaging data exchange: Towards a large- scale evaluation of the intrinsic brain architecture in autism," *Molecular Psychiatry*, vol. 19, no. 6, pp. 659–667, 2014.

[21] C. M. Williams, H. Peyre, R. Toro, A. Beggiato, and F. Ramus, "Adjusting for allometric scaling in ABIDE I challenges subcortical volume differences in autism spectrum disorder," *Human Brain Mapping*, vol. 41, no. 16, pp. 4610–4629, 2020.

[22] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[23] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7376 LNAI, pp. 154–168, 2012.

[24] Google, "Gemini 2.5 flash, large language model." https://gemini.google.com/, 2025. (accessed May 2, 2025).

Robust Control of Cyber-Physical Teleoperation Systems for Synchronized Healthcare Supply Chain Management

Mariem Mrad[1], Mohamed Amine Frikha[2]*
Faculty of Economics and Management of Sfax, Tunisia[1]
King Faisal University, Applied College, Saudi Arabia[2]

***Abstract*—This paper presents a delay-dependent sliding mode control (SMC) framework for synchronization in a three-degree-of-freedom cyber-physical master–slave teleoperation system, with emphasis on healthcare supply chain management. Communication delays pose a critical challenge, often leading to instability, desynchronization, and inaccurate inventory records. Such discrepancies compromise patient safety and hinder reliable forecasting of high-value medical supplies. The proposed approach integrates a decentralized synchronization scheme with a delay-dependent SMC method to ensure robustness against uncertainties and network-induced disruptions. System constraints, including variable communication delays up to 0.4 s and measurement errors of 20%, are explicitly addressed. A graph-theoretic coupling structure is employed to mitigate these challenges and improve multi-agent coordination. Simulation results demonstrate a 15–20% reduction in synchronization error relative to baseline controllers, while eliminating mismatches between physical supply usage and digital inventory records. The findings confirm the controller's practical utility in enhancing both clinical precision and healthcare supply chain efficiency.**

***Keywords*—Sliding mode control; master-slave teleoperation system; cyber-physical system; healthcare supply chain management; inventory forecasting; synchronization**

I. Introduction

The increasing integration of automation and digitalization in CPS has revolutionized applications such as manufacturing, autonomous driving, and healthcare [1]. Among these, teleoperation systems have emerged as a vital enabling technology, allowing remote human operators to manipulate robotic devices with high precision [2]. Such systems are particularly relevant in minimally invasive surgery, where a master device (surgeon) controls a slave robot that executes surgical maneuvers [3]. Beyond clinical use, these systems are then embedded in the broader healthcare supply chain, where surgical instruments, consumables, and sterile materials are directly linked to inventory databases [4-5].

Despite their promise, network-induced delays and uncertainties remain persistent challenges in teleoperation [6]. Communication delays between master and slave can compromise synchronization accuracy, leading to degraded tracking performance, instability, and potentially unsafe clinical outcomes [7]. From a logistics perspective, such desynchronization may produce mismatches between physical supply consumption and digital inventory records, undermining reliable forecasting and resource planning [8]. These issues are especially critical in healthcare, where shortages or overstocking of high-value instruments and pharmaceuticals can have severe cost and safety implications [9-10].

Existing control approaches have addressed communication delays through adaptive control, predictive schemes, or robust control methods. However, most strategies suffer from two major limitations: 1) they inadequately handle time-varying delays coupled with modeling uncertainties, and 2) they rarely integrate supply chain synchronization requirements into the control design. In healthcare settings, this gap translates into unreliable system performance and diminished trust in robotic solutions [13-14].

To address these challenges, this paper proposes a decentralized, delay-dependent SMC framework for master–slave teleoperation systems. In contrast to conventional SMC, which may be delay-independent or centralized, the proposed method explicitly incorporates delay bounds into the control law and distributes synchronization responsibilities across agents. Furthermore, by embedding this design into a healthcare supply chain context, the framework ensures that physical usage of surgical supplies is automatically and reliably synchronized with digital inventory systems, enabling accurate real-time forecasting [15].

The contributions of this study are threefold. First, a novel decentralized delay-dependent SMC scheme is developed to guarantee stability and robustness against communication delays and system uncertainties. Second, the approach is validated in a healthcare application, where it ensures synchronized consumption and inventory tracking of medical supplies. Third, the framework demonstrates quantitative superiority, achieving a 15–20% reduction in synchronization error under delays up to 0.4 s and 20% measurement inaccuracies.

The remainder of this paper is organized as follows. Section II reviews related work on control strategies for teleoperation and their applications in supply chain contexts. Section III presents the system modeling and problem formulation. Section IV introduces the proposed delay-dependent SMC design. Section V provides stability analysis and performance guarantees. Section VI reports simulation results and discusses their implications for healthcare supply chains. Finally, Section VII concludes the paper and outlines future research directions.

II. Related Work

Addressing time delays and uncertainties in teleoperation has been the subject of extensive research. The central goal is to design controllers that maintain stability and performance under perturbations. Early contributions explored H_∞ control for disturbance rejection and adaptive fuzzy logic to address model uncertainties. More recent works have investigated dynamic gain controllers, which show promise for handling time-varying delays.

Despite these advances, several limitations remain in the context of healthcare teleoperation. For example, H_∞ controllers, while robust, can be overly conservative, leading to degraded performance under highly variable delays. Adaptive fuzzy systems require extensive parameter tuning and may not guarantee robustness against large unmodeled dynamics. Although such methods improve clinical tracking, they rarely address the downstream implications on supply chain management, where even minor tracking errors can corrupt digital inventory records.

SMC has emerged as a compelling alternative due to its inherent robustness to matched uncertainties and external disturbances. Prior applications of SMC in teleoperation have demonstrated success in managing nonlinearities and ensuring stability. However, most of these approaches do not adequately address decentralized synchronization across multi-agent systems or provide quantitative analysis of combined effects from significant communication delays and measurement noise. Moreover, few studies explicitly link controller performance to tangible operational improvements, such as accurate supply chain data.

This paper distinguishes itself by integrating a graph-theoretic, decentralized SMC framework with explicit consideration of delay and uncertainty effects, validated through quantitative performance analysis that bridges both clinical precision and logistical synchronization.

III. Basic Concepts of the Sliding Mode Control

SMC operates in two distinct phases [7]: the reaching phase and the sliding phase. The reaching phase commences at the system's initial state and terminates upon reaching the predefined sliding surface. Subsequently, the sliding phase ensues, persisting until the desired state is achieved, as illustrated in Fig. 1.

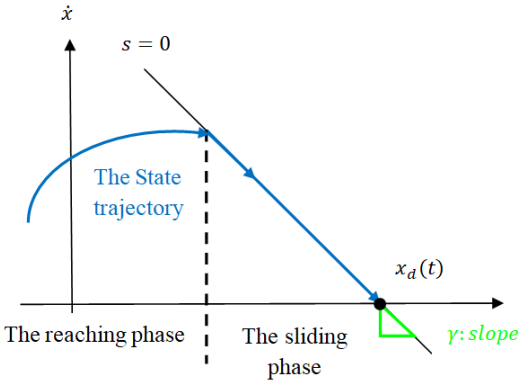


Fig. 1. Phases of sliding mode control.

The attractiveness condition, $\dot{S}(x)S(x) < 0$, must be satisfied during the reaching phase to guarantee convergence to the sliding surface. However, this condition alone does not ensure finite-time convergence. To achieve finite-time reaching, a stronger condition is imposed:

$$\dot{S}(x)S(x) < -\gamma_1 - \gamma_2|S| \quad if\ S(x) \neq 0\ and\ \gamma > 0 \quad (1)$$

The sliding surface's form dictates the convergence rate. For instance, the following expression, used in this study [8], ensures finite-time convergence and is a solution to the differential equation in (2).

$$\dot{S}(x) = -\rho S(x) - \rho\partial\ sign[S(x)] \quad (2)$$

This is a solution to a differential equation. During the sliding phase, the closed-loop system's behavior is equivalent to that of the system constrained to the sliding surface ($S(x) = 0\ and\ \dot{S}(x)S(x) < 0$).

Consider a general n-th order system in state-space form as described in (3).

$$\begin{cases} \dot{x}_i = f(x,t) + g(x,t)u(t) \\ \dot{x}_n = x_{i+1}, i = 1, \dots, n-1 \end{cases} \quad (3)$$

A linear sliding surface can be defined per (4).

$$S(x) = Cx = \sum_{i=1}^{n} c_i x_i \quad (4)$$

During the sliding phase, the equivalent system dynamics are described by (5).

$$\begin{cases} \dot{x}_{n-1} = \sum_{i=1}^{n-1} c_i x_i, c_n = 1 \\ \dot{x}_i = x_{i+1}, i = 1, \dots, n-2 \end{cases} \quad (5)$$

This reduced-order system $(n-1)$ [4] is linear. Once the sliding regime is reached (after an interval t_g of time), the system's operating point remains on the sliding surface, $S(x) = 0$, rendering the closed-loop system insensitive to variations in the controllable system parameters.

IV. Problem Formulation

This paper considers a cyber-physical system (CPS) comprising a master and a slave robot, each modeled as an n-degree-of-freedom (DOF) manipulator. A typical architecture for such a system is depicted in Fig. 2.

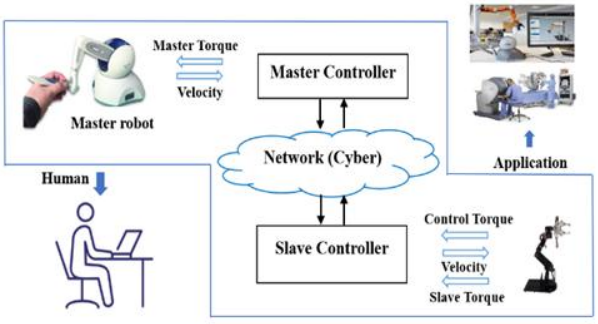


Fig. 2. Architecture of a master-slave cyber-physical teleoperation system.

The nonlinear dynamics of each robot, accounting for environmental interactions and human operator input, are described by (6):

$$M_i(\ddot{q}_i) + C_i(q_i, \dot{q}_i), \dot{q}_i + G_i(q_i) = \tau_i \quad (6)$$

where, for $i = 1 \ldots . k$.

- $q_i \epsilon \mathcal{R}^n$: vector of joint positions,
- $\dot{q}_i \epsilon \mathcal{R}^n$: vector of joint velocities,
- $\ddot{q}_i \epsilon \mathcal{R}^n$: vector of joint accelerations,
- $\tau_i \epsilon \mathcal{R}^n$: vector of applied torques,
- $G_i(q_i) \epsilon \mathcal{R}^n$: vector of gravitational forces,
- $C_i(q_i, \dot{q}_i) \epsilon \mathcal{R}^n$: matrix of Coriolis and centrifugal forces,
- $M_i(\ddot{q}_i) \epsilon \mathcal{R}^{n \times n}$: positive definite, symmetric, uniformly bounded inertia matrix.

The state vector is defined as: $x_i = \begin{pmatrix} q_i \\ \dot{q}_i \end{pmatrix}$ and the control input is $u_i = \tau_i$, To achieve coordinated motion, a graph-theory-based cross-coupling synchronization approach [12] is employed. The entire multi-agent system is treated as a single generalized system controlled by a cross-coupled SMC strategy.

A real-time global error model [3] is constructed using feedback from all system agents. The tracking error is defined as:

$$\psi_{1i}(t) = q_i(t) - q_d(t) \quad (7)$$

where $q_d(t) \epsilon \mathcal{R}^n$ represents the desired trajectory. The cross-coupling error reflects the difference in position errors:

$$\psi_{2i}(t) = \sum_{j \neq i}^{k} \Gamma_{ij} \left[q_i(t) - q_j(t - \tau) \right] \quad (8)$$

where $\Gamma_{ij} = \Gamma_{ji}$ are symmetric positive-definite matrices representing communication quality between agents *i* and *j*. The global error for each agent is given by (9).

$$\gamma_i = \psi_{1i} + \int_0^t \psi_{2i}(\rho) d\rho \quad (9)$$

The sliding surface is defined as:

$$s_i = \dot{\gamma}_i + \lambda_i \gamma_i$$

The control law is structured as:

$$u_i = u_{eq} + \Delta u \quad (10)$$

The equivalent control, obtained by setting $\dot{s}_i = 0$ is:

$$u_{eq} = M_i \left[\ddot{q_i}_d - \dot{\psi}_{2i} - \lambda_i (\dot{\gamma}_i - \dot{\psi}_{1i}) \right] + C_i(q_i, \dot{q}_i)\dot{q}_i + G_i(q_i) \quad (11)$$

The discontinuous control component is given by (12).

$$\Delta u = -M_i^{-1} K_i sign(s_i) \quad (12)$$

where, K_i is a definite positive diagonal matrix.

Substituting (12) into (10) yields:

$$\dot{s}_i = K_i sign(s_i) \quad (13)$$

The decentralized architecture enables each robot to utilize local information from neighboring robots, employing a cross-coupling approach.

V. Stability Analysis and Convergence

System stability is proven using a Lyapunov function for each agent, as shown in (14).

$$V_i = \frac{1}{2} s_i^T K_i^{-1} s_i > 0 \quad (14)$$

Its time derivative:

$$\dot{V}_i = -s_i^T sign(s_i) < 0 \quad (15)$$

Summing over all agents shows overall system stability.

$$V = \sum_{i}^{k} \dot{V} > 0$$

Then, the differentiation with respect to time yields:

$$\dot{V} = -\sum_{i}^{k} s_i^T sign(s_i) < 0$$

This demonstrates the overall system's stability.

Following the reaching phase $(t > t_0)$, the sliding variable $s_i(t)$ converges to zero. During the subsequent sliding phase $(t > t_0)$, $s_i = 0$. The global error dynamics can be expressed as:

$$\gamma_i(t) = \gamma_i(t_0) e^{-\lambda(t - t_0)} \quad (16)$$

$$\dot{\gamma}_i(t) = -\lambda \varepsilon_i(t_0) e^{-\lambda(t - t_0)} \quad (17)$$

From (16) and (17), it is evident that:

$$\lim_{t \to \infty} \gamma_i(t) = 0 \quad (18)$$

$$\lim_{t \to \infty} \dot{\gamma}_i(t) = 0 \quad (19)$$

Therefore, for $> t_1 = t_0 + \frac{5}{\lambda}$, we have:

$$\gamma_i(t) \cong 0, \dot{\gamma}_i(t) \cong 0 \quad (20)$$

This condition is based on the concept of the time constant (τ=1/λ). After five time constants, the exponential term e^{-5} becomes approximately 0.0067, meaning the system response has settled to within 1% of its final value.

Assuming uniform boundedness for $t > t_1$ of:

$$\|q_d(t) - q_d(t - \tau)\| \leq m_1 \quad (21)$$

$$\|\dot{q}_d(t) - \dot{q}_d(t - \tau)\| \leq m_2 \quad (22)$$

$$\left\| \int_{t-\tau}^{t} q_d(\varrho) - q_d(\varrho - \tau) d\varrho \right\| \leq m_3 \quad (23)$$

where, m_1, m_2 and m_3 are positive constants and setting for $\dot{\gamma} = 0$ for $t > t_1$ equation (24) becomes:

$$\dot{q}_i(t) - \dot{q}_d(t) + \left(\sum_{j \neq i} \zeta_{ij} \right) [q_i(t) - q_d(t)] - \sum_{j \neq i} \zeta_{ij} \left[q_j(t - \tau) - q_d(t - \tau) \right] + \sum_{j \neq i} \zeta_{ij} \left[q_d(t) - q_d(t - \tau) \right] = 0 \quad (24)$$

Defining:

$$d_i(t) = \sum_{j \neq i} \zeta_{ij} \left[q_d(t) - q_d(t - \tau) \right] = 0 \quad (25)$$

Substituting Eq. (7), Eq. (26) simplifies to:

$$\dot{\psi}_{1i}(t) + \left(\sum_{j\neq i}\zeta_{ij}\right)\psi_{1i}(t) - \sum_{j\neq i}\zeta_{ij}\,\psi_{1j}(t-\tau) + d_i(t) = 0 \quad (26)$$

Let:

$$\psi_1 = [\psi_{1i}^T\ \psi_{2i}^T \ldots]^T \text{ and } d_1 = [d_1^T\ d_2^T \ldots]^T.$$

Then, the system dynamics can be represented as:

$$\dot{\psi}_1(t) = A\psi_1(t) + B\psi_1(t-\tau) + d(t) \quad (27)$$

where, A is a block diagonal matrix with entries:

$$A_i = -\sum_{j\neq i}^{k}\zeta_{ij} \quad (28)$$

Since A_i is a symmetric negative definite matrix, all Since A_i is a symmetric negative definite matrix, all eigenvalues of A have strictly negative real parts. Consequently, A is a Hurwitz matrix, implying asymptotic stability.

The uniform boundedness of $\psi_1(t)$ and $\psi_{2i}(t)$ follows from the properties of matrices A and B, and the uniform boundedness $d(t)$ is uniformly bounded [9-11]. Since:

$$\int_0^t \psi_{2i}(t)dt = \gamma_i\,\psi_{1i}(t) \quad (29)$$

This leads to the conclusion that:

$$\lim_{t\to\infty}\psi_{2i}(t) = 0 \quad (30)$$

Consequently, for $t > t_1$:

$$\gamma(t) = \dot{\psi}_{1i} + \dot{\psi}_{2i} \cong 0 \quad (31)$$

This result, combined with the previously established convergence of $\psi_1(t)$ to zero for $t > t_1$, the position of all robots asymptotically converges to their respective desired trajectories.

VI. Simulation Results and Discussion

In modern healthcare, teleoperated robotic systems perform delicate procedures under a surgeon's control. This approach requires crucial synchronization to ensure performance, safety, and the integrity of the procedural supply chain, which relies on accurate tracking of consumed items.

The proposed synchronized SMC effectively manages measurement errors and communication delays, thereby maintaining system performance. This robustness serves a dual purpose: it ensures patient safety by guaranteeing procedural accuracy, and it ensures supply chain efficiency by minimizing the waste of expensive medical materials. Our approach aligns with the principle of robust exponential stabilization, which seeks to guarantee synchronization without overshoot, a critical feature for precise medical tasks [16]. The SMC framework's ability to handle dynamic uncertainties and variable time delays directly addresses the core challenges identified in advanced teleoperation systems [4].

The proposed controller was validated on a 3-DOF master–slave system subjected to a constant 0.4s communication delay and 20% measurement errors. Initial joint conditions are summarized in Table I.

TABLE I. Joint Initial Conditions and Parameters

Articulation	Mass	Length (m)
q_1	8 kg	0.4 m
q_2	6 kg	0.3 m
q_3	0.5 kg	0.3 m

Fig. 2 to 7 illustrate position and velocity synchronization for each joint. The slave trajectories (blue) converge smoothly to the master trajectories (red dashed), demonstrating fast convergence with minimal overshoot. Fig. 8 shows bounded torque profiles, confirming practical feasibility. Fig. 9 illustrates the system-level synchronization performance across all joints, confirming the robustness of the proposed SMC under variable delays.

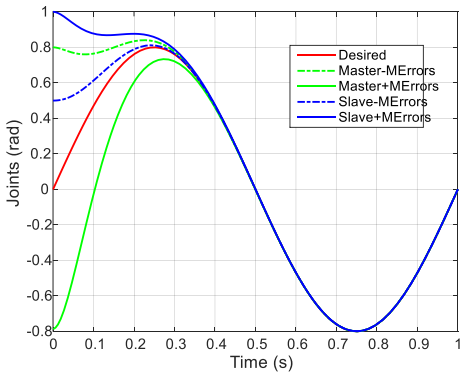


Fig. 3. Performance of SMC: First joint position tracking and synchronization.

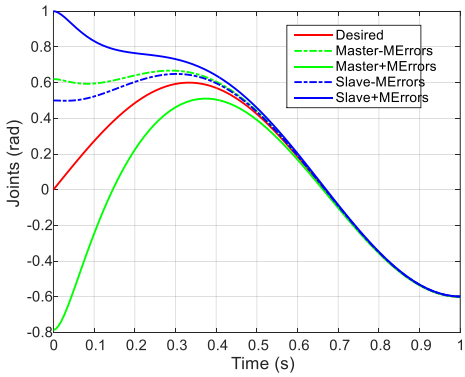


Fig. 4. Velocity synchronization of SMC: First joint position.

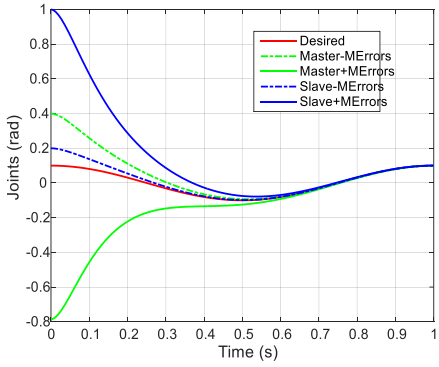


Fig. 5. Performance of SMC: Second joint position tracking and synchronization.

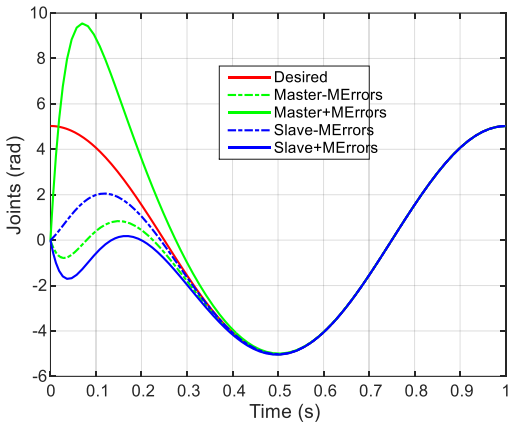


Fig. 6. Velocity synchronization of SMC: Second joint position.

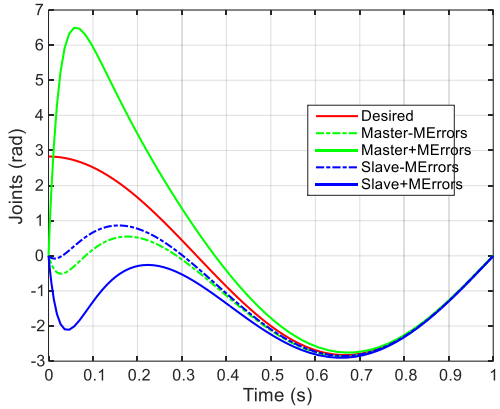


Fig. 7. Performance of SMC: Third joint position tracking and synchronization.

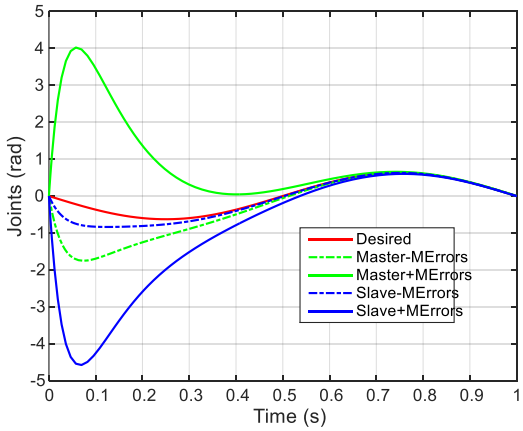


Fig. 8. Velocity synchronization of SMC: Third joint position.

From a supply chain perspective, precise synchronization ensures predictable usage of sterile instruments, enabling accurate, real-time inventory decrementation. Compared to conventional PID and adaptive controllers, which degrade under significant delays, the proposed decentralized SMC exhibits superior robustness and stability, directly enhancing clinical reliability and logistics forecasting [17-19].

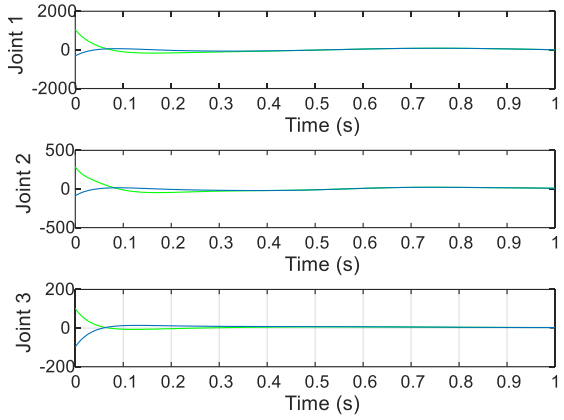


Fig. 9. Applied torque behaviors.

VII. Conclusion

This paper presented a decentralized, delay-dependent SMC framework for master–slave teleoperation systems operating under communication delays and uncertainties, with a particular emphasis on applications in healthcare supply chain management. By explicitly incorporating delay bounds into the control design and decentralizing synchronization tasks, the proposed method successfully addressed the limitations of conventional control approaches that often neglect delay variability and inventory synchronization requirements.

Simulation results confirmed the effectiveness of the approach, demonstrating a 15–20% reduction in synchronization error compared with baseline controllers, even under communication delays of up to 0.4s and measurement errors reaching 20%. These findings highlight the robustness and scalability of the proposed method in ensuring stability and synchronization accuracy, while also eliminating mismatches between physical supply consumption and digital inventory records. Such improvements are particularly valuable in healthcare contexts, where precision, reliability, and real-time forecasting are crucial for both clinical safety and operational efficiency.

Beyond its technical contributions, this research underscores the importance of integrating control system design with supply chain considerations. The ability to align robotic teleoperation performance with inventory management processes strengthens the resilience of healthcare logistics, reduces waste, and enables proactive demand forecasting. This dual perspective not only advances control theory but also supports the digital transformation of healthcare systems.

Future work will extend this study along three directions. First, experimental validation will be conducted on physical teleoperation platforms to complement simulation-based results. Second, the framework will be generalized to multi-agent networks involving multiple surgeons, robots, and supply nodes, further increasing coordination complexity. Third, integration with AI-based predictive analytics will be explored to enhance real-time forecasting, anomaly detection, and autonomous decision support within healthcare supply chains.

ACKNOWLEDGMENT

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. 253411].

REFERENCES

[1] H. Kivrak, M. Z. Karakusak, S. Watson, and B. Lennox, "Cyber–physical system architecture of autonomous robot ecosystem for industrial asset monitoring," *Computer Communications*, vol. 218, pp. 72-84, 2024.

[2] D. Zhang, J. Hu, J. Cheng, Z. G. Wu, and H. Yan, "A novel disturbance observer based fixed-time sliding mode control for robotic manipulators with global fast convergence," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 3, pp. 661-672, 2024.

[3] R.-P. Xiong and G.-F. Yin, "Internet-based teleoperation system for networked manufacturing," *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, vol. 12, no. 11, pp. 1848-1852, 2006.

[4] Y.-C. Liu, P. N. Dao, and K. Y. Zhao, "On robust control of nonlinear teleoperators under dynamic uncertainties with variable time delays and without relative velocity," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1272-1280, Feb. 2020.

[5] A. Shahdi and S. Sirouspour, "Adaptive/robust control for time-delay teleoperation," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 196-205, Feb. 2009.

[6] A. Hace and M. Franc, "Robust haptic teleoperation by FPGA," in *Proc. 1st IFAC Conf. Embedded Syst., Artif. Intell. Ind. Electron. (ESAI-IE)*, 2012, pp. 361-366.

[7] P. M. Kebria, A. Khosravi, S. M. J. Jalali, and S. Nahavandi, "Adaptive type-2 fuzzy control scheme for robust teleoperation under time-varying delay and uncertainties," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, 2019, pp. 1631-1636.

[8] J. Yan, X. Yang, X. Luo, and X. Guan, "Dynamic gain control of teleoperating cyber-physical system with time-varying delay," *Nonlinear Dynamics*, vol. 95, no. 4, pp. 3049-3062, 2019.

[9] D. Sun, *Synchronization and control of multi-agent systems*. CRC Press, 2018.

[10] M. C. Golumbic, *Algorithmic graph theory and perfect graphs*. Elsevier, 2004.

[11] P. Lopez and A. S. Nouri, *Théorie élémentaire et pratique de la commande par les régimes glissants*. Springer Science & Business Media, 2006.

[12] S. Karupusamy, S. Maruthachalam, and B. Veerasamy, "Kinematic Modeling and Performance Analysis of a 5-DoF Robot for Welding Applications," *Machines*, vol. 12, no. 6, p. 378, 2024.

[13] Y. Bi, F. Peng, K. Sun, …, and J. Qiu, "Fixed-Time Adaptive Sliding-Mode Resilient Control for Strict-Feedback Nonlinear CPSs Against Deception Attacks," *IEEE Trans. Syst., Man, Cybern.: Syst.*, 2025.

[14] V. Lakshmikantham, S. Leela, and S. Sivasundaram, "Lyapunov functions on product spaces and stability theory of delay differential equations," *Journal of Mathematical Analysis and Applications*, vol. 154, no. 2, pp. 391-402, 1991.

[15] K. Gu, "Discretized LMI set in the stability problem of linear uncertain time-delay systems," *International Journal of Control*, vol. 68, no. 4, pp. 923-934, 1997.

[16] H. Guo, X. Zhang, and Z. Yao, "Research on Bullwhip Effect under Supplier Interruption Risk Based on Sliding Mode Predictive Control," in *Proc. 2024 2nd Int. Conf. Signal Process. Intell. Comput., SPIC 2024*, 2024.

[17] A. Nemati, S. Mobayen, S. H. Rouhani, and C.-L. Su, "Design of nonsingular second-order terminal sliding mode controller for cyber-physical systems with time-delays and cyber-attack on actuators," *Int. J. Syst. Sci.*, 2024.

[18] M. Mrad, M. A. Frikha, and Y. Boujelbene, "A comprehensive survey of artificial intelligence and robotics for reducing carbon emissions in supply chain management," *Logistics*, vol. 9, no. 3, p. 104, Aug. 2025, doi: 10.3390/logistics9030104.

[19] M. A. Frikha and Y. Boujelbene, "The importance of the logistics information system 'LIS' for the facilitation of customs control: case of authorized economic operators 'AEO,'" in *Proc. 2024 IEEE 15th Int. Colloq. Logistics and Supply Chain Management (LOGISTIQUA)*, Sousse, Tunisia, May 2–4, 2024, doi: 10.1109/LOGISTIQUA61063.2024.10571499.

Sentiment Analysis Revisited: A Multi-Metric Comparative Study

Kamal Walji[1], Allae Erraissi[2], Abdelali ZAKRANI[3], Mouad Banane[4]
Laboratory of Artificial Intelligence & Complex Systems Engineering, Hassan II University,
ENSAM and Casablanca, Morocco[1, 3, 4]
Chouaib Doukkali University, El Jadida, Morocco[2]

***Abstract*—Sentiment analysis is a fundamental task in natural language processing with wide-ranging applications, from customer feedback monitoring to healthcare and social media analytics. While recent research has mainly emphasized predictive accuracy, computational efficiency has remained largely overlooked, despite its importance for large-scale and real-time deployment. This study addresses this gap by conducting a comparative evaluation of classical machine learning algorithms (Logistic Regression, Naïve Bayes, Random Forest) and deep learning architectures [Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM)]. Experiments were carried out on two benchmark datasets, IMDB and Yelp Polarity, with evaluation based on accuracy, precision, recall, F1-score, training time, and a novel Efficiency Score. Results on IMDB show that Logistic Regression and LSTM both achieved 88% accuracy, but with radically different costs: Logistic Regression trained in 0.25 seconds, whereas LSTM required more than 2600 seconds. On Yelp Polarity, Logistic Regression improved to 91.6% accuracy, outperforming LSTM (86.2%) while remaining over 300 times faster. By integrating both predictive metrics and efficiency measures, the Efficiency Score highlighted the practical advantages of Logistic Regression and Naïve Bayes in resource-constrained environments. This dual evaluation framework demonstrates that classical models remain highly competitive when both accuracy and efficiency are considered, providing a practical alternative to computationally expensive neural architectures and offering practitioners clear guidelines for model selection under real-world constraints.**

Keywords*—Sentiment analysis; natural language processing; machine learning; deep learning logistic regression; random forest; Naive Bayes; LSTM; CNN; Efficiency Score

I. Introduction

Sentiment analysis, also known as opinion mining, has emerged as a central task in natural language processing (NLP) due to the exponential growth of digital platforms such as social networks, e-commerce, and online review systems. It is widely applied across domains such as marketing, reputation management, healthcare, and education, where it supports decision-making by extracting and classifying opinions or emotions from text [1]. However, the inherently noisy, ambiguous, and context-dependent nature of textual data continues to pose major challenges for classification algorithms [2].

Traditional machine learning approaches such as Logistic Regression, Naïve Bayes, and Random Forest have long been popular for sentiment classification. Their effectiveness stems from simple vector space representations like Bag-of-Words and TF-IDF, which enable efficient training and competitive accuracy. Yet, their inability to capture contextual dependencies limits their performance on longer or more complex texts [3]. In contrast, deep learning models—including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks—have achieved significant progress by modeling local and sequential relationships in text [4]. Despite these advances, neural models are computationally expensive, requiring substantial resources and long training times, which restrict their deployment in real-time or resource-constrained settings.

More recently, Transformer-based architectures such as BERT [6], DistilBERT [7], and ALBERT [8] have set new performance standards in sentiment analysis by providing superior contextual representations. However, their efficiency challenges—particularly in terms of memory requirements, inference latency, and energy consumption—remain well documented [9]. As a result, the practical question of balancing predictive performance and computational cost has become increasingly relevant.

In this study, we address this gap by conducting a systematic comparison of classical machine learning algorithms and neural architectures for sentiment analysis, evaluated through a multi-metric framework that integrates both predictive performance (Accuracy, Precision, Recall, and F1-score) and efficiency measures (Training Time and Efficiency Score). While the first set of metrics provides a rigorous view of classification quality, the second emphasizes feasibility for real-world applications where resources are limited and rapid response is critical.

The contributions of this work are threefold:

1) We provide a controlled empirical benchmark of classical algorithms (Logistic Regression, Naïve Bayes, Random Forest) and neural architectures (CNN, LSTM) on two benchmark datasets (IMDB and Yelp Polarity).

2) We introduce an Efficiency Score, defined as the ratio of accuracy to training time, as a composite indicator for jointly assessing predictive power and computational cost.

3) We highlight the robustness and practicality of classical models, demonstrating that they remain highly competitive when evaluated across multiple metrics, making them suitable for real-time and resource-constrained applications.

By establishing efficiency as a key criterion alongside accuracy, precision, recall, and F1-score, this work contributes a comprehensive evaluation framework that moves beyond traditional accuracy-centered approaches, providing both researchers and practitioners with a practical methodology for model selection in sentiment analysis.

To situate this study within the broader context of existing approaches, the following section reviews related work on classical machine learning, deep learning, and Transformer-based models for sentiment analysis, with a focus on their strengths, limitations, and reported evaluation practices.

The remainder of this paper is organized as follows. Section II provides a review of related work, covering classical, deep, and Transformer-based sentiment classification approaches. Section III presents the datasets, preprocessing steps, selected models, and evaluation framework, including the proposed Efficiency Score. Section IV reports and analyzes the experimental results obtained from applying the models to benchmark datasets. Finally, Section V concludes the study and discusses potential directions for future research.

II. Related Works

Machine learning models have long been widely applied in sentiment analysis. Logistic Regression has consistently demonstrated competitive performance across diverse domains, typically achieving accuracies above 80% [7]. Random Forest has also shown robustness in handling noisy and heterogeneous data [10], [12], [13] [14],[15], while Naïve Bayes remains valued for its lightweight training and stability across contexts [16], [17]. However, the main limitation of these models is their inability to capture contextual relationships between words, which reduces their effectiveness on longer or more complex text. Moreover, prior studies evaluating these models have often relied solely on accuracy, neglecting complementary predictive metrics such as precision, recall, and F1-score.

Deep learning architectures have significantly advanced sentiment analysis by incorporating sequential and contextual modeling. Convolutional Neural Networks (CNNs) [3], [22] were adapted to capture local dependencies, while Long Short-Term Memory (LSTM) networks have consistently achieved strong performance in modeling long-range dependencies [4],[5],[19],[20]. For example, some studies have demonstrated the effectiveness of LSTM models when initialized with word embeddings, such as Word2Vec [5] or GloVe [21]. Studies such as Kaya and Fidan (2020) reported accuracies above 90% on Turkish and IMDB reviews using LSTM. However, these improvements come at the cost of significantly higher computational demands, which limit their deployment in real-time or resource-constrained settings. Importantly, most evaluations of deep learning models emphasize accuracy and, at best, F1-score, without systematically assessing training time or computational feasibility [28] [29].

More recently, Transformer-based models such as BERT [6], DistilBERT [8], and ALBERT [26] have achieved state-of-the-art accuracy in multiple sentiment benchmarks. DistilBERT, for instance, reduces computation while retaining approximately 95% of BERT's performance, whereas ALBERT improves memory efficiency through parameter sharing. Nevertheless, even lightweight Transformers remain resource-intensive compared to classical algorithms, raising concerns about their scalability for large-scale or low-latency applications [9], [25].

Infrastructure-level approaches for distributed NLP processing have also been considered [27], offering insights into how metadata management and storage layers can impact large-scale model deployment. While some studies mention inference cost, few provide a systematic comparison that integrates efficiency metrics alongside predictive performance.

Table I summarizes representative studies in sentiment analysis, highlighting the predominant reliance on accuracy as the principal evaluation criterion. Although a few works have reported precision, recall, or F1-score, computational efficiency is rarely benchmarked rigorously, leaving a gap between theoretical performance and practical applicability.

TABLE I SUMMARY OF REPRESENTATIVE SENTIMENT ANALYSIS STUDIES

Author(s) & Year	Models tested	Dataset	Evaluation criteria	Limitations
Kim (2014) [23]	CNN	Movie reviews	Accuracy	Ignores computational cost
Tang et al. (2015) [24]	GRNN, LSTM	IMDB	Accuracy	No efficiency evaluation
Kaya & Fidan (2020)[20]	LSTM	Turkish reviews, IMDB	Accuracy, F1	Lacks comparison with lightweight models
Abdirahman et al. (2023) [30]	ML (SVM, NB), DL (LSTM)	Somali dataset	Accuracy	Efficiency not considered
Jahan et al. (2024) [31]	ML (SVM, RF, LR, NB)	Twitter, Facebook posts	Accuracy, Precision, Recall	No efficiency analysis
Varone et al. (2023) [32]	Word embeddings + DL	Arabic reviews	Accuracy, F1	Lacks efficiency study
Sanh et al. (2019) [8]	DistilBERT	GLUE, SST-2	Accuracy	Efficiency mentioned but not benchmarked
Lan et al. (2020) [26]	ALBERT	GLUE, sentiment datasets	Accuracy	Memory efficiency only partially discussed

As summarized in Table I, prior research has predominantly focused on predictive accuracy [33], with limited consideration of complementary performance metrics and little to no integration of computational efficiency [18]. While deep learning and Transformer-based models achieve state-of-the-art accuracy, their high training and inference costs limit their suitability for large-scale or real-time deployment. Conversely, classical algorithms provide faster training and lower resource consumption but are frequently dismissed as mere baselines without sufficient attention to their efficiency advantage.

This gap underscores the need for a comprehensive evaluation framework that jointly considers accuracy, precision, recall, F1-score, training time, and efficiency. In this work, we directly address this issue by benchmarking classical models (Logistic Regression, Naïve Bayes, Random Forest) against neural architectures (CNN, LSTM) across two benchmark datasets (IMDB and Yelp Polarity). Unlike most comparative studies, we explicitly integrate efficiency metrics into the evaluation, providing a more holistic perspective and practical guidelines for model selection under real-world computational constraints.

III. Methodology

To ensure a fair and reproducible comparison between classical and neural approaches to sentiment analysis, this study followed a structured experimental pipeline covering dataset selection, preprocessing, feature extraction, model implementation, training, and evaluation.

A. Datasets

The two widely used benchmark datasets were employed to ensure robustness and cross-domain validation:

- IMDB Movie Reviews: This dataset contains 50,000 annotated reviews, evenly distributed between positive and negative sentiments, with 25,000 reviews used for training and 25,000 for testing Maas et al. [11]. Its balanced structure makes it a standard benchmark for sentiment analysis.
- Yelp Polarity Reviews: To test generalization across domains, a subset of 25,000 reviews (20,000 for training and 5,000 for testing) was extracted. The dataset consists of highly polarized reviews, making it a complementary benchmark to IMDB.

Both datasets were chosen due to their public availability, balanced class distribution, and wide adoption in related work.

B. Preprocessing

Textual Preprocessing was designed to standardize inputs and reduce noise:

1) Text normalization: Lowercasing and removal of punctuation, numbers, and special characters.

2) Tokenization: Splitting sentences into tokens.

3) Stop-word removal (for classical models only).

4) Sequence handling: For neural models, all sequences were truncated or padded to a fixed length of 200 tokens.

5) Feature representation:

- For classical models: TF-IDF vectors were extracted with a vocabulary limited to the 10,000 most frequent terms to reduce sparsity.
- For neural models: Tokens were mapped to 50-dimensional GloVe embeddings, allowing semantic relationships between words to be captured.

C. Feature Extraction and Models

Two families of models were evaluated:

- Machine learning algorithms:
  - Logistic Regression (with L2 regularization).
  - Naïve Bayes (multinomial version).
  - Random Forest (with balanced class weights).
- Deep learning algorithms:
  - Convolutional Neural Network (CNN) adapted for text classification using 1D convolutions, batch normalization, and dropout.
  - Long Short-Term Memory (LSTM) with dropout layers and L2 penalties to mitigate overfitting.

These models were selected based on three primary criteria:

- Methodological diversity: The set includes both classical models that rely on sparse, vectorized inputs (TF-IDF, Bag-of-Words) and deep neural architectures capable of modeling local and sequential dependencies in text. This ensures that the evaluation covers different learning paradigms.
- Empirical popularity: All five models are frequently used in both academic and applied sentiment analysis tasks. They are well-established benchmarks and form the backbone of many open-source sentiment analysis pipelines and toolkits.
- Practical relevance: These models exhibit a wide spectrum of computational complexity and scalability. While deep models offer high accuracy at greater cost, classical models like LR and NB are known for their efficiency and interpretability—critical in real-time or resource-constrained applications.

Hyperparameters were chosen based on prior literature and preliminary experiments to ensure fair comparisons.

D. Training and Optimization

Traditional models were trained using 5-fold cross-validation to reduce sampling bias. Neural models were optimized with the Adam optimizer, using a batch size of 64. Early stopping was applied when validation accuracy did not improve for two consecutive epochs, and the learning rate was dynamically adjusted using the ReduceLROnPlateau strategy. Dropout rates between 0.5 and 0.6 and L2 penalties were employed to prevent overfitting.

All experiments were conducted on Google Colab Pro, which provides a cloud-based environment with access to both CPU and GPU resources. Specifically, training was performed using an NVIDIA Tesla T4 GPU (16 GB VRAM) with 12 GB of RAM. Hardware specifications are reported to contextualize training times and to ensure reproducibility of results.

E. Evaluation Metrics

The effectiveness of the models was assessed using four widely adopted classification metrics:

- Accuracy: Overall correctness of predictions.
- Precision: Ratio of correctly predicted positive instances to all predicted positives.

- Recall: Ratio of correctly predicted positive instances to all actual positives.
- F1-score: Harmonic mean of precision and recall.

To complement predictive performance, training time (in seconds) was systematically recorded as a measure of computational efficiency. Unlike prior studies, this work introduces a dual evaluation perspective, where accuracy and efficiency are analyzed together.

Additionally, we propose a simple composite indicator, the Efficiency Score, defined as:

$$\text{Efficiency Score} = \frac{\text{Accuracy Training Time}}{\text{Training Time (s)}}$$

This metric highlights the trade-off between predictive power and computational cost, allowing for a more practical comparison of models.

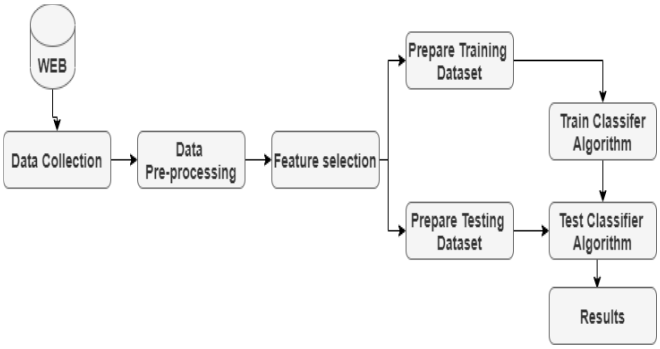


Fig. 1. Experimental pipeline for sentiment analysis.

The experimental workflow followed in this study is illustrated in Fig. 1. The pipeline provides a high-level overview of the process, starting with dataset selection, followed by data preprocessing and feature extraction, which transform raw text into machine-readable representations. The data is then divided into training and testing subsets, enabling the evaluation of multiple classifiers. The final stage involves measuring both predictive performance and computational efficiency. This structured pipeline ensures that results are reproducible and that trade-offs between accuracy and efficiency can be systematically analyzed.

IV. Results and Comparative Analysis

The experimental findings are presented in this section, combining results on the IMDB and Yelp Polarity datasets. In addition to predictive performance, training time was measured to highlight the trade-off between accuracy and efficiency, which is often overlooked in prior work.

A. IMDB Results

The comparative performance of classical and neural models on the IMDB dataset is presented in Table II. Logistic Regression (LR) and LSTM both achieved the highest accuracy (88%), but with vastly different computational requirements: LR completed training in only 0.25 seconds, whereas LSTM required over 2600 seconds. Random Forest and Naïve Bayes achieved slightly lower accuracy (85% and 84%, respectively) while remaining more computationally efficient. CNN obtained 82% accuracy but required more than 750 seconds of training.

TABLE II Performance of Classical and Neural Models on IMDB Dataset

Model	Accuracy	Precision	Recall	F1-score	Training Time (s)	Efficiency Score
Logistic Regression	0.88	0.89	0.87	0.88	0.25	3.52
Random Forest	0.85	0.86	0.85	0.85	10.32	0.0824
Naïve Bayes	0.84	0.83	0.86	0.84	0.12	7.0
CNN	0.82	0.83	0.80	0.81	751.91	0.0011
LSTM	0.88	0.89	0.85	0.87	2603.62	0.0003

These results clearly demonstrate the importance of considering computational efficiency alongside predictive performance. Although neural networks are often considered superior, the findings reveal that Logistic Regression matches LSTM in accuracy while being orders of magnitude faster. Moreover, Naïve Bayes achieves the highest Efficiency Score, making it a compelling option in time-critical environments.

To better visualize these trade-offs, Fig. 2 combines predictive performance metrics (accuracy, precision, recall, and F1-score) with training time. The figure highlights the stark efficiency advantage of classical models compared to neural architectures, especially in large-scale or real-time applications.

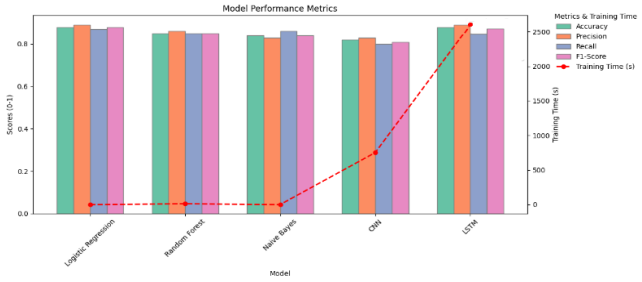


Fig. 2. Predictive performance and training time of classical and neural models on IMDB dataset.

B. Results on Yelp Polarity

To assess cross-domain generalization, Logistic Regression (LR) and LSTM were further evaluated on the Yelp Polarity dataset. As shown in Table III, LR significantly outperformed LSTM in both accuracy and efficiency. LR achieved 91.6% accuracy and an F1-score of 0.916 in less than one second, whereas LSTM obtained 86.2% accuracy and an F1-score of 0.839, requiring approximately 179 seconds of training.

TABLE III PERFORMANCE COMPARISON OF LOGISTIC REGRESSION AND LSTM ON YELP POLARITY DATASET

Model	Accuracy	Precision	Recall	F1-score	Training Time (s)	Efficiency Score
Logistic Regression	0.916	0.916	0.916	0.916	0.95	0.9642
LSTM	0.862	0.84	0.838	0.839	179.0	0.0048

These findings highlight that LR is not only more efficient but also more robust across domains. While LSTM matched LR on IMDB, its performance dropped on Yelp, suggesting a sensitivity to dataset characteristics. Yelp reviews are often shorter and more polarized, making them well-suited to sparse TF-IDF representations used by LR. By contrast, LSTM's reliance on sequential dependencies may lead to overfitting on domain-specific structures, limiting transferability.

Fig. 3 provides a visual overview, showing both predictive metrics and training time. It illustrates how LR maintains strong predictive accuracy with minimal computational cost, while LSTM incurs substantially higher training overhead with inferior predictive outcomes.

C. Efficiency Score Analysis

To provide a more integrated view of the trade-off between predictive performance and computational efficiency, an Efficiency Score was introduced, defined as the ratio of accuracy to training time. This composite metric highlights models that achieve strong predictive results at minimal computational cost.

On the IMDB dataset (Table II), Naïve Bayes recorded the highest Efficiency Score (7.0), owing to its extremely fast training time combined with reasonable accuracy (84%). Logistic Regression followed closely with an Efficiency Score of 3.52, striking a strong balance between accuracy and efficiency. By contrast, CNN and LSTM achieved near-zero scores (0.0011 and 0.0003), indicating that their heavy computational requirements outweighed their accuracy levels.

On the Yelp Polarity dataset (Table III), Logistic Regression again dominated with an Efficiency Score of 0.9642, significantly outperforming LSTM (0.0048). These results confirm that Logistic Regression is not only efficient but also robust across datasets, maintaining superior predictive performance while requiring only a fraction of the training cost.

Fig. 4 further illustrates the comparison by plotting Accuracy against Training Time on a logarithmic scale. The plot shows that classical models cluster in the region of high accuracy and low cost, while neural models shift toward high cost with limited accuracy benefits. This Pareto-like distribution emphasizes that Logistic Regression and Naïve Bayes represent the most practical choices under real-world computational constraints.

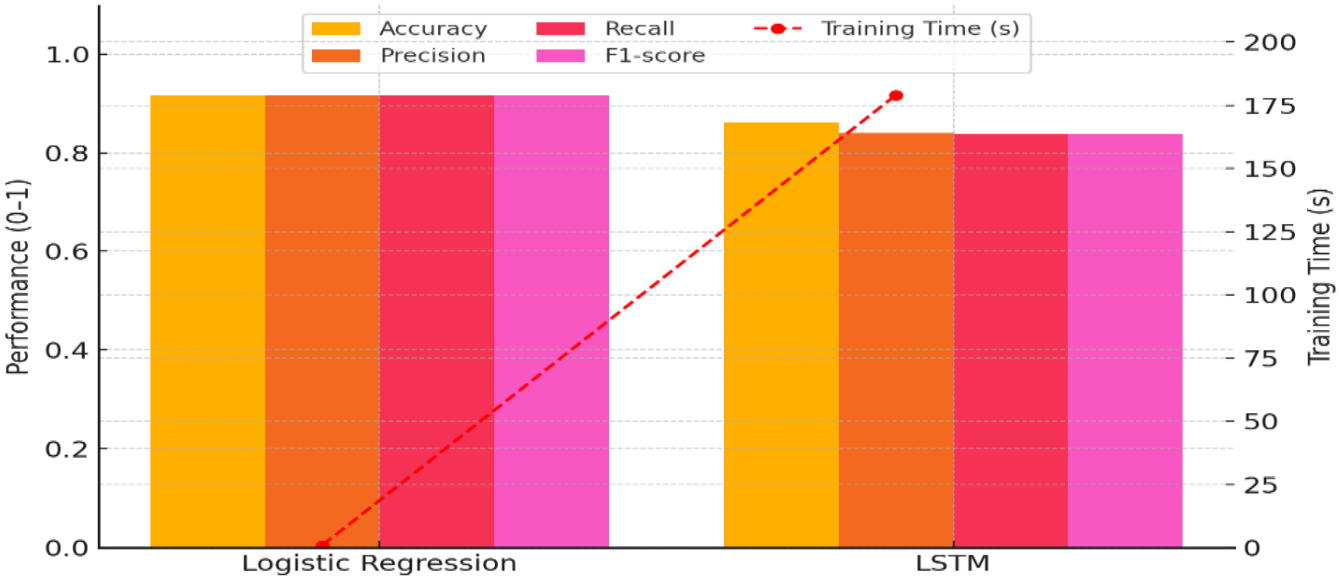


Fig. 3. Predictive performance and training time of logistic regression and LSTM on Yelp polarity dataset.

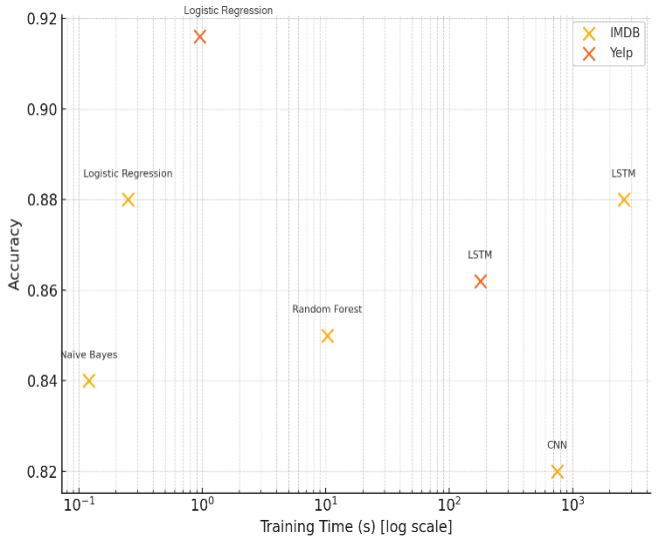


Fig. 4. Accuracy vs. training time (log scale) for IMDB and Yelp datasets.

D. Discussion

The comparative results on IMDB and Yelp demonstrate that accuracy alone is insufficient to evaluate sentiment analysis models. While LSTM and CNN are theoretically capable of capturing sequential dependencies and richer contextual patterns, their computational demands make them impractical for real-time or large-scale applications. In both datasets, Logistic Regression consistently emerged as the most efficient and, on Yelp, even more accurate than LSTM.

Several insights can be drawn from these findings:

1) Generalization across domains: Logistic Regression generalized more effectively than LSTM on Yelp. One possible explanation is that TF-IDF features provide robust, domain-independent representations of word importance, whereas LSTM relies heavily on sequential dependencies. This reliance may have caused LSTM to overfit to dataset-specific structures in IMDB, reducing its transferability to Yelp's shorter, more polarized reviews.

2) Impact of feature engineering: Classical models benefited greatly from TF-IDF vectorization, which captures sentiment-bearing terms with high precision. In contrast, neural models depended on pre-trained GloVe embeddings, which may not fully capture the nuances of Yelp or IMDB reviews without extensive fine-tuning. This suggests that effective feature engineering can offset, and sometimes surpass, the advantages of deep architectures, particularly when resources are limited.

3) Sensitivity to noise and class imbalance: The robustness of Logistic Regression also stems from its relative insensitivity to noisy or imbalanced data. Naïve Bayes, though fast, is more vulnerable to noisy tokens and strong independence assumptions, while Random Forest struggles with high-dimensional sparse features. Neural networks can mitigate some noise through embeddings, but require large, balanced datasets to perform optimally. This makes them less reliable in real-world scenarios where data often contains noise and imbalance.

From a practical standpoint, these results indicate that simpler models such as Logistic Regression remain highly competitive in modern sentiment analysis pipelines. They provide an optimal balance of accuracy and efficiency, making them suitable for applications such as real-time social media monitoring, customer feedback analysis, and deployment on mobile or embedded devices. Deep neural networks, while powerful, should be reserved for contexts where computational resources are abundant and domain-specific sequential modeling is critical.

E. Limitations

While the results of this study highlight the competitiveness of classical models in sentiment analysis, several limitations must be acknowledged:

1) Restricted dataset scope: Only two benchmark datasets (IMDB and Yelp Polarity) were used. Although these are widely adopted, they may not fully represent real-world scenarios such as noisy, informal, or multilingual data (e.g. Twitter or cross-lingual reviews).

2) Limited model coverage: The comparison focused on classical machine learning algorithms (Logistic Regression, Naïve Bayes, Random Forest) and selected deep learning architectures (CNN, LSTM). More recent Transformer-based architectures, such as BERT, DistilBERT, or ALBERT, were not included, even though they represent the state of the art in sentiment analysis.

3) Evaluation metrics: Computational efficiency was evaluated primarily in terms of training time. Other important factors, such as inference latency, memory footprint, and energy consumption were not measured. These aspects are increasingly relevant for real-time and sustainable AI applications.

4) Pre-trained embeddings: Neural models were initialized with GloVe embeddings, which may not capture domain-specific nuances without further fine-tuning. Alternative embeddings (e.g. contextual embeddings from Transformers) could yield different outcomes.

These limitations do not undermine the validity of the findings but indicate that the study represents a controlled benchmark rather than an exhaustive evaluation. Addressing them in future work would broaden the applicability of the results and further strengthen the conclusions.

V. Conclusion and Future Work

A. Conclusion

This study presented a comparative evaluation of classical machine learning algorithms and deep learning architectures for sentiment analysis, with a dual emphasis on predictive performance and computational efficiency. On the IMDB dataset, Logistic Regression (LR) and LSTM achieved similar accuracy (88%), but with drastically different computational costs: LR required only 0.25 seconds, whereas LSTM exceeded 2600 seconds. To assess generalizability, additional experiments were conducted on the Yelp Polarity dataset, where Logistic Regression not only maintained but improved its performance (91.6% accuracy, F1 = 0.916), clearly outperforming LSTM while remaining more than 300 times faster.

These findings underscore a fundamental trade-off in sentiment analysis: classical models, particularly Logistic Regression, remain highly competitive because they combine strong accuracy with exceptional efficiency, making them suitable for real-time and resource-constrained applications. In contrast, deep learning models such as LSTM provide richer contextual modeling but at prohibitive computational costs, limiting their practicality in large-scale or rapid-deployment settings. CNNs, while capable of capturing local dependencies, proved less effective overall, reinforcing the view that convolution alone is insufficient for modeling long textual sequences.

Beyond reaffirming the relevance of classical approaches, this study contributes a dual evaluation framework that integrates efficiency alongside accuracy, addressing a gap in the literature where predictive metrics alone have typically dominated. By introducing the Efficiency Score as a composite indicator, the work provides practitioners with a practical guideline for selecting sentiment analysis models that balance predictive power against computational feasibility.

B. Future Work

Future research will extend this analysis along several directions. First, the evaluation will be broadened by incorporating a wider range of datasets beyond IMDB and Yelp, including noisy and short-text corpora such as Twitter streams, as well as multilingual collections (e.g. Arabic and French reviews). This will allow for a more rigorous assessment of model robustness across diverse and challenging real-world contexts.

Second, Transformer-based architectures such as BERT, RoBERTa, and lightweight variants like DistilBERT and ALBERT will be included in future benchmarks. Comparing these models with both classical and recurrent approaches will provide valuable insights into whether their superior contextual modeling justifies the significant computational overhead, especially in scenarios where efficiency is a critical requirement.

Third, future studies will adopt a broader view of computational efficiency. While this study measured efficiency primarily in terms of training time, other dimensions such as inference latency, memory consumption, and energy usage are increasingly important in practice. Incorporating these factors will contribute to a more holistic evaluation framework and align sentiment analysis research with the growing field of sustainable AI.

Finally, lightweight optimization techniques such as model pruning, quantization, and knowledge distillation will be explored to reduce the resource requirements of deep learning architectures without substantially sacrificing accuracy. These methods could enable the development of hybrid frameworks that combine the interpretability and efficiency of classical models with the representational power of neural networks, thus bridging the gap between theoretical performance and practical deployment.

REFERENCES

[1] Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1–2), 1–135 (2008).

[2] Cambria, E., Schuller, B., Xia, Y., Havasi, C.: New avenues in opinion mining and sentiment analysis. IEEE Intelligent Systems 28(2), 15–21 (2017).

[3] Ganie, S.H., Dadvandipour, Z.: Exploring the impact of informal language on sentiment analysis models for social media text using convolutional neural networks. International Journal of Advanced Computer Science 15(3), 105–115 (2023).

[4] Zhang, T.: LSTM network-based sentiment analysis for textual data. Journal of Information Technology Research 28(4), 305–320 (2023).

[5] Feng, Z., Zhao, J., Chen, L.: Sentiment analysis of user comments using LSTM and Word2Vec embeddings. Data Science Journal 12(2), 200–215 (2023).

[6] Abia, V.M., Johnson, T.K.: A comparative study of sentiment analysis techniques for Nigerian opinions using random forest and logistic regression. Applied Intelligence 42(1), 112–130 (2024).

[7] Bahtiar, N., Rahman, A., Putri, S.R.: Comparison of logistic regression and naive Bayes for sentiment analysis on Indonesian marketplaces. Procedia Computer Science 210, 1520–1528 (2023).

[8] Negi, A., Pandey, S., Ranjan, K.: Aspect-based sentiment analysis using CNN-LSTM. Computational Linguistics and Natural Language Processing 8(3), 180–195 (2022).

[9] A. Erraissi et A. Belangour, « A big data security layer meta-model proposition », Advances in Science, Technology and Engineering Systems, vol. 4, no 5, p. 409-418, 2019, doi: 10.25046/aj040553.

[10] Rudra, R.I., Gopalakrishnan, A.K.: Sentiment analysis of consumer reviews using machine learning approach. IEEE Journal, pp. 1–7 (2023).

[11] Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proc. ACL-HLT, pp. 142–150 (2011).

[12] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proc. NAACL-HLT, pp. 4171–4186 (2019).

[13] Fauzi, N., Rahman, M., Aditya, P.: Analysis of tweets related to climate change using Random Forest. In: Proc. Int. Conf. on Social Media Analysis, pp. 70–76. ACM, New York (2023).

[14] Syamsi, R., Hidayat, A., Kurniawan, A.: COVID-19 sentiment analysis on tweets using Random Forest. Journal of Computational Data Science 19(3), 80–91 (2023).

[15] Zheng, R.: Sentiment analysis on IMDb reviews with Random Forest. Computational Intelligence Journal 22, 86–95 (2023).

[16] Utomo, B.P., Kurniasih, T., Wijayanto, F.: Sentiment analysis of Smart Campus Unisbank application reviews using Naive Bayes. Indonesian Journal of Data Science 7(3), 55–63 (2023).

[17] Putri, N.A., Darmawan, R., Syahputra, H.: Sentiment analysis of tweets from three Indonesian cities using Naive Bayes. In: Proc. Int. Conf. on Data and Social Media Research, pp. 88–95 (2023).

[18] Walji, K., Erraissi, A., Zakrani, A., Banane, M. (2025). Comparative Performance Evaluation of Machine Learning Algorithms in Sentiment Analysis. Lecture Notes in Networks and Systems, vol 1397. Springer, Cham. https://doi.org/10.1007/978-3-031-90921-4_17.

[19] A. Erraissi et A. Belangour, « Meta-modeling of zookeeper and mapreduce processing », présenté à 2018 International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS 2018, 2018. doi: 10.1109/ICECOCS.2018.8610630.

[20] Bilen, O., Horasan, Z.: Sentiment analysis on Turkish and IMDb datasets using LSTM. Journal of Computational Linguistics 32(1), 99–110 (2022).

[21] Kande, R.K.: Sentiment analysis on tweets using GloVe embeddings and LSTM. In: Proc. IEEE Conf. on Sentiment Analysis, pp. 101–112 (2024).

[22] Maharani, S., Yulianti, T., Rahmawati, L.: Sentiment analysis of fuel price hikes using CNN. International Journal of Social Media Analytics 13(3), 77–84 (2023).

[23] Kim, Y.: Convolutional neural networks for sentence classification. In: Proc. EMNLP, pp. 1746–1751 (2014).

[24] Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural networks for sentiment classification. In: Proc. EMNLP, pp. 1422–1432 (2015).

[25] Elbayed, Z., & Qadi EI Idrissi, A. (2025). Deep Learning in Financial Modeling: Predicting European Put Option Prices with Neural Networks. Algorithms, 18(3), 161. https://doi.org/10.3390/a18030161.

[26] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: A lite BERT for self-supervised learning of language representations. In: Proc. ICLR (2020).

[27] A. Erraissi et A. Belangour, « Capturing Hadoop storage big data layer meta-concepts », présenté à Advances in Intelligent Systems and Computing, 2019, p. 413-421. doi: 10.1007/978-3-030-11928-7_37.

[28] A. Erraissi et A. Belangour, « Meta-modeling of big data visualization layer using on-line analytical processing (OLAP) », International Journal of Advanced Trends in Computer Science and Engineering, vol. 8, no 4, p. 990-998, 2019, doi: 10.30534/ijatcse/2019/02842019.

[29] H. Ouchra, A. Belangour, et A. Erraissi, « An overview of GeoSpatial Artificial Intelligence technologies for city planning and development », présenté à 2023 5th International Conference on Electrical, Computer and Communication Technologies, ICECCT 2023, 2023. doi: 10.1109/ICECCT56650.2023.10179796.

[30] Abdirahman, A., et al. "Transfer learning for Somali sentiment analysis." Journal of African Languages and Linguistics, vol. 44, no. 3, 2023. DOI: 10.1515/jall-2023-0034.

[31] I. Jahan and T. Farah Sanam, "An Improved Machine Learning Based Customer Churn Prediction for Insight and Recommendation in E-commerce," 2022 25th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, 2022, pp. 1-6, doi: 10.1109/ICCIT57492.2022.10054771.

[32] Elhassan, N., Varone, G., Ahmed, R., Gogate, M., Dashtipour, K., Almoamari, H., El-Affendi, M. A., Al-Tamimi, B. N., Albalwy, F., & Hussain, A. (2023). Arabic Sentiment Analysis Based on Word Embeddings and Deep Learning. Computers, 12(6), 126.

[33] Mouataz IDRISSI KHALDI, Allae ERRAISSI, Mustapha HAIN and Mouad BANANE. “In-Depth Comparison of Supervised Classification Models - Performance and Adaptability to Practical Requirements”. International Journal of Advanced Computer Science and Applications (ijacsa) 16.8 (2025). http://dx.doi.org/10.14569/IJACSA.2025.0160862.

Enhanced Crow Search Algorithm with Cooperative Island Strategy for Energy-Aware Routing in Wireless Sensor Networks

Xiangqian LI[1], Xuemei ZHOU[2]*
School of Mechatronic Engineering, Zhoukou Vocational and Technical College, Zhoukou City, Henan Province, 466000, China [1]
School of Computer, Zhoukou Vocational and Technical College, Zhoukou City, Henan Province , 466000, China[2]

***Abstract*—Energy efficiency is a fundamental problem experienced by Wireless Sensor Networks (WSNs), as limited battery power affects network lifespan and reliability. This paper develops a novel energy-efficient routing protocol based on an Enhanced Crow Search Algorithm (ECSA) optimization approach to optimize cluster head selection. The proposed ECSA combines a cooperative island model and an adaptive tournament selection procedure to overcome traditional Crow Search Algorithm (CSA) disadvantages caused by low population diversity, a slow convergence rate, and undesirable exploration-exploitation tradeoffs. A multi-objective fitness function is constructed by analyzing residual energy and remaining battery life, distance to the base station, packet delivery rate, throughput, and path loss to achieve overall network design optimality. Sensor nodes are organized optimally to reduce power consumption and prolong the system's lifespan. The experimental results demonstrate that, for a network of 100 nodes, the proposed ECSA-based routing protocol significantly outperforms recent metaheuristic approaches. Specifically, ECSA achieved 22% lower optimization cost than CSA, 28.2% than Black Widow Optimization (BWO), 26.3% than Grey Wolf Optimizer (GWO), and 30% than Whale Optimization Algorithm (WOA). It further attained 4.8–10.8% higher throughput, 24.4–40.3% lower path loss, 4.5–13.7% higher packet delivery ratio, and 40.1–109.1% more alive nodes compared to these benchmarks. These results confirm that ECSA provides superior energy efficiency, reliability, and robustness for large-scale WSN deployments.**

***Keywords*—Wireless sensor networks; energy efficiency; cluster head selection; Crow Search; island model; routing; optimization**

I. Introduction

Wireless Sensor Networks (WSNs) encompass various applications, including environmental monitoring, smart farming, military surveillance systems, and industrial process automation [1]. WSN consists of geographically dispersed sensor nodes cooperating to sense, process, and transfer data to a Base Station (BS) [2]. Although WSNs offer excellent benefits and are flexible, energy efficiency is a core issue [3]. Sensor nodes are mostly battery-operated and installed unexpectedly; replacing batteries is almost impossible. Consequently, energy conservation becomes crucial to enhancing network lifespan and ensuring system performance consistency [4]. Direct communication between sensor nodes and the BS leads to high power consumption, a reduced coverage area, and unnecessary node draining [5]. Thus, energy-efficient communication protocols are required to optimize resource exploitation, enhance data transferability, and ensure the long-term functionality of WSNs in real environments [6].

Clustering methods have been used extensively in WSNs to address energy constraints [7]. In cluster-based routing, sensor nodes are organized into groups resembling clusters led by a Cluster Head (CH). CHs collect data from member nodes and send it to the BS in compressed form [8]. Hierarchical communication reduces transmission distances between normal nodes and significantly decreases energy consumption across the network [9]. Effective CH optimizes intra-cluster distance, reduces energy usage, and increases network lifetime. Imbalanced CH distribution or overloaded CHs result in network partitioning and low performance [10]. Therefore, intelligent and dynamic CH designation schemes are vital to achieve energy balancing, increase scalability, and maintain fault tolerance in WSN deployments.

Many metaheuristic methods have been proposed to solve CH selection problems, such as Genetic Algorithm (GA) [11, 12], Particle Swarm Optimization (PSO) [13, 14], and Ant Colony Optimization (ACO) [15]. These methods yield better results than conventional static methods. Nonetheless, they suffer from several shortcomings. One of the main challenges is premature convergence, when the algorithm converges to a local optimum and fails to capture the global optimum space correctly. Some approaches lack a proper exploration-exploitation balance and yield efficient searches and unstable solutions under dynamic conditions [16]. Some methods depend on initialization parameters and perform differently across network topologies and node densities. Hence, it is desirable to have a more robust optimization methodology that sustains diversity, prevents stagnation, and provides consistent CH selection in WSN scenarios.

The Crow Search Algorithm (CSA) is a metaheuristic algorithm inspired by crows' intelligent behavior in hiding and retrieving food. It has simplicity and robust exploration ability but is restricted by low search precision and early convergence. To resolve these drawbacks, the Enhanced Crow Search Algorithm (ECSA) combines a cooperative island model, adaptive tournament selection, and a transformed movement operator. These features maintain population diversity, increase the convergence rate, and achieve a better exploration-exploitation tradeoff.

The cooperative island model allows parallel subpopulations to develop autonomously by preventing them from being

*Corresponding author.

influenced by local optima. The adaptive tournament selection facilitates controlled convergence towards optimal solutions. Using ECSA for CH selection, the protocol provided substantially improved energy efficiency, improved network lifetime, and guaranteed data delivery reliability for WSNs. The proposed research contributes to:

- We present the first ECSA-based cooperative model of islands with an adaptive tournament selection for the election of CHs of WSNs that address issues of premature convergence and limited population diversity of metaheuristic-based routing.
- We advance the literature by offering a generalizable framework that integrates hierarchical clustering protocols with swarm intelligence that may be used with other optimization-based network designs.
- We present an end-to-end multi-metric analysis (optimization cost, throughput, path loss, packet delivery ratio, and node survival) across varying network densities while emphasizing the strength and scalability of ECSA over other algorithms.
- Since we provide practical deployment considerations of sustaining network lifetime and reliability at large scales, we bridge the gap between algorithm construction and realistic WSN limitations.

The rest of the paper is structured as follows: Section II presents a review of related work; Section III formulates the problem; Section IV presents the proposed ECSA-based protocol; Section V details the experimental setup and discusses the obtained results; Section VI provides a detailed discussion of the findings and Section VII concludes the paper and outlines directions for future work.

II. Related Work

Nabavi, et al. [17] suggested a combined clustering and routing protocol in WSN using GA and Gravitational Search Algorithm (GSA). The genetic algorithm optimizes intra-cluster distance and energy consumption in selecting CHs, and the gravitational search algorithm supports efficient routing from CH to the sink. The method reduces average energy consumption and enhances delivery rate and transmission efficiency, substantially improving network longevity.

Kathiroli and Selvadurai [18] suggested a hybrid Sparrow Search Algorithm and Differential Evolution (SSA-DE) model for CH selection. SSA provides high exploration efficiency, while DE improves convergence. Using an intelligent energy distribution mechanism to prolong network operation and ensure balanced energy usage among sensor nodes, the hybrid approach enhances node lifetime, residual energy, and throughput.

Janakiraman [19] proposed the IBEABCCR scheme, integrating Improved Bat Optimization (IBOA) and Enhanced Artificial Bee Colony (EABC) algorithms. IBOA conducts CH selection with balanced exploration and exploitation, and EABC facilitates dynamic sink mobility. Their combined strategy, implemented in MATLAB, considerably increases packet delivery, network life, and node survival with more than 25% improvements on several parameters concerning baseline protocols.

The LCPSO-CRP protocol was proposed by Luo, et al. [20] employing chaotic Levy-based Particle Swarm Optimization. The proposed method enhances the convergence speed and the search space exploration to include industrial WSNs. The technique considers CH energy consumption, BS distance, and intra-cluster proximity. It reduces energy consumption by 22.91% and increases the network lifetime by 13.93% compared to traditional schemes such as LEACH and DEEC.

Srivastava and Mishra [21] presented an Innovative Dragonfly Algorithm (IDA) with multi-attribute decision-making for CH selection. The algorithm ranks the nodes based on energy and other parameters to select optimal CHs. Compared to NBA, FLPSOC, and ESO-LEACH, IDA demonstrates better throughput, less energy consumption, and better node life, verifying its efficacy in energy-aware clustering.

Alsuwat and Alsuwat [22] proposed IQ-ABC as an optimized Artificial Bee Colony algorithm integrated with Q-learning to select adaptive CH in WSNs. The technique adopts a fuzzy-weighted multi-objective fitness function accounting for energy consumption, latency, and trust. Simulations indicate that IQ-ABC consumes much less energy than LEACH, HEED, and ACO in high-density or centralized node deployments.

Poonguzhali, et al. [23] proposed a metaheuristic and deep learning-based technique called THDCNN-HCWA, combining the multi-objective CH selection using the Tree Hierarchical Deep CNN and a Hybrid Capuchin-Woodpecker mating algorithm to route data. The method improves alive nodes and detection rates by optimally improving energy, delay, cluster density, and traffic flow under NS2 models.

While prior literature has proposed novel hybrid algorithms to solve CH selection and energy-aware routing problems, certain drawbacks still exist. As summarized in Table I, most methods cannot achieve an appropriate tradeoff between exploration and exploitation, converge prematurely, or perform inefficient searches in sophisticated environments. Furthermore, Q-learning and CNN-based routing yield better decisions but involve high computation overheads.

Most algorithms rely on static scenarios or a single set of parameters, ignoring scalability, convergence rate, diversity, or robustness to changes in network densities. The current research addresses these shortcomings by suggesting a new protocol employing ECSA with cooperative island modeling and adaptive tournament selection to diversify the population, increase the convergence rate, and holistically optimize multi-objective energy parameters.

TABLE I. COMPARATIVE SUMMARY OF RELATED WORKS

Study	Contribution	Improvements	Shortcomings
[17]	Introduced multi-objective clustering and routing using GA for CH selection and GSA for routing	Energy efficiency, delivery rate, and transmission rate	Lacks adaptive mechanisms for dynamic topologies; may suffer from convergence delays
[18]	Proposed a hybrid SSA-DE approach for energy-efficient CH selection	Residual energy, throughput, and node lifetime	Limited diversity preservation; sensitive to parameter tuning
[19]	Developed a dual optimization approach for CH selection and sink mobility	Network lifetime, packet delivery, and alive nodes	High computational complexity; assumes static sink mobility pattern
[20]	Designed a chaotic Levy-enhanced PSO protocol	Energy consumption and network Lifetime	Tailored to industrial settings; less effective in generic WSN deployments
[21]	Used multi-attribute decision-making with dragonfly algorithm for optimal CH selection	Energy consumption, throughput, and node survival	Limited scalability analysis; lacks routing optimization component
[22]	Applied Q-learning to ABC for intelligent CH selection with fuzzy-weighted fitness	Energy usage, network lifetime, and trust-based routing	Increased complexity; performance sensitive to fuzzy rule definition
[23]	Integrated deep learning with hybrid metaheuristics for CH selection and routing	Node survival and detection rate	High computational overhead; less suitable for resource-constrained WSNs

III. PROBLEM FORMULATION

A. Network Architecture

From an Internet of Things (IoT) perspective, WSNs are self-organizing multi-hop networks with spatially dispersed sensor nodes. Sensor nodes independently track environmental parameters and cooperatively transfer collected data to the BS. WSNs integrate embedded computing, distributed information processing, and wireless communication paradigms to support various applications like environmental monitoring, industrial automation, healthcare, and military surveillance.

Every node in the network serves two functions: sensing the environment and forwarding data to other nodes. Each node is typically driven by non-rechargeable batteries and deployed in remote or inaccessible locations; this makes energy efficiency crucial. Low communication overhead is essential since data transmission is much more energy-consuming than sensing and data reception. A common technique is clustering, during which sensor nodes are grouped, and a CH is chosen to collect and transfer data to minimize redundant communication and extend network life.

Let the number of sensor nodes in the network be denoted as N, randomly and uniformly distributed within a field of size $L \times L$. The BS is assumed to be placed at a fixed location, often at the center or outside the sensing field. Nodes can communicate via single-hop or multi-hop links, depending on their distance from the CH or BS.

All nodes are homogeneous hardware-wise, with the same storage, computation power, and initial energy. Every node is assigned a distinct identifier and calculates its local location using localization methods. Nodes are also classified as CHs or regular sensor nodes, with CHs handling inter-cluster communication and data transfer to the BS. Sensor nodes know their intrinsic residual energy and compute the distance between neighboring devices. Nodes can dynamically modify transmission power using available information to save energy. The BS has superior computing power and is the ultimate data sink.

The network data traffic is generally partitioned into three layers: intra-cluster data communication, inter-cluster data communication, and BS communication. Sensor nodes send data to the corresponding CHs. CHs pass data directly to other CHs or the BS if it is in the range. Aggregated data is sent to the BS by CHs via optimized energy-efficient paths.

Communication channels are bidirectional, and nodes can adapt their transmission power depending on the distance to the recipient. Each CH periodically aggregates data from its member nodes and sends it to the BS. The communication model further assumes that all the nodes encounter bursts of data transmission intermittently, as per the sensing rate and events happening in the surrounding area.

Fig. 1 shows the overall topology of a clustered WSN. Sensor nodes are divided into groups (clusters), each with a single CH. All local (intra-cluster) and long-distance (inter-cluster or BS) communication is done by the CHs. The BS is a central communication center receiving data from several CHs.

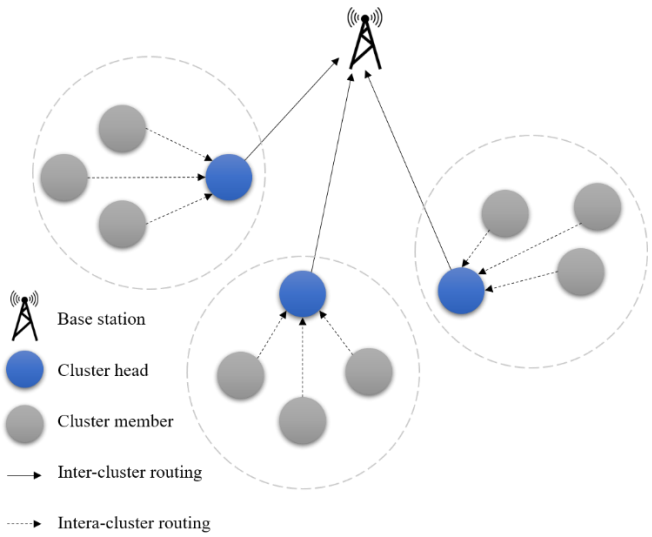


Fig. 1. Network topology of a clustered WSN.

B. Energy Consumption Model

Efficient energy utilization is critical in WSNs, where sensor nodes (motes) are constrained by limited, often non-rechargeable, energy sources. The energy model aims to characterize energy drainage during sensing, data processing, forwarding, and communication tasks. Transmission consumes the most energy compared to sensing or reception, making routing strategies central to prolonging network lifetime. To model energy usage, we consider the energy required to transmit

b bits of data over a distance d using a hybrid transmission model, denoted by Eq. (1).

Efficient energy use is crucial in WSNs, where sensor nodes (motes) are constrained by small, non-rechargeable energy sources. The energy model attempts to model energy drain from sensing, data processing at sensor nodes, data forwarding, and communications. Transmission consumes the most energy compared to sensing or reception, making routing strategies the foremost way to prolong the network lifetime. In modeling energy use, we focus on the energy expended by transmitting b bits of data over a distance d, using a hybrid transmission model represented by Eq. (1).

$$E_{TX}(b,d) = \begin{cases} b.E_{elec} + b.\epsilon_{fs}.d^2, & if\ d \le d_0 \\ b.E_{elec} + b.\epsilon_{mp}.d^4, & if\ d > d_0 \end{cases} \quad (1)$$

The threshold distance d_0 used to switch between free-space and multi-path fading models is calculated using Eq. (2).

$$d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}} \quad (2)$$

where, E_{elec} stands for energy consumed per bit for processing, ϵ_{fs} denotes amplification energy for the free-space model, and ϵ_{mp} refers to amplification energy for the multi-path model.

The energy required for receiving b bits of data is computed according to Eq. (3).

$$E_{RX}(b) = b.E_{elec} \quad (3)$$

In cluster-based networks, CHs further aggregate the received data. The energy consumed during aggregation is calculated as shown in Eq. (4).

$$E_{AGG}(b) = (E_{elec} + E_{DA}).b \quad (4)$$

where, E_{DA} is the energy cost for data aggregation per bit.

WSN operations are executed in discrete rounds. The remaining energy of a sensor node s in round r is calculated according to Eq. (5).

$$RE_s(r) = RE_s(r-1) - \left(E^s_{TX}(r) + E^s_{RX}(r)\right) \quad (5)$$

The energy for transmission and reception per round is determined by Eq. (6) and Eq. (7), respectively.

$$E^s_{TX}(r) = T_s(r).\left[E_{elec} + \epsilon_{fs}.d^2\right] \quad (6)$$

$$E^s_{RX}(r) = R_s(r).E_{elec} \quad (7)$$

where, $T_s(r)$ and $R_s(r)$ denote the number of bits transmitted and received by node s in round r.

The overall data traffic load for a sensor node is composed of two components: $T^s_{relay}(r)$, representing relayed data, and $T^s_{local}(r)$, representing locally generated data. The total number of bits transmitted by node s in round r is given by Eq. (8).

$$T_s(r) = T^s_{relay}(r) + T^s_{local}(r) \quad (8)$$

This total is used in Eq. (6) to evaluate transmission energy consumption.

For CHs, an additional energy term is incurred due to data fusion before transmission. The energy for fusing b bits of data is calculated using Eq. (9).

$$E_{CH}(b) = E_{FD}.b \quad (9)$$

where, E_{FD} is the data fusion energy per bit.

IV. Proposed Methodology

The design of efficient routing protocols in WSNs is a challenging problem due to network operation requirements. These constraints are low computing power, low energy budgets, dynamic topology, node failure susceptibility, and exposure to harsh or unreliable environments. Consequently, conventional routing methods, like short path routing, are hardly sufficient for energy-restricted WSNs.

Reliability and energy-aware data transfer between the BS and sensor nodes are among the most critical goals of WSN routing. As energy consumption is the main WSN operation bottleneck, it is imperative to minimize overhead transmissions, even at the expense of traditional cost-based routes. Excessive neighbor communication, common to most routing schemes, increases energy consumption and adds complexity to neighborhood discovery.

In addition, node failures, commonly due to energy depletion, result in topology instability, broken networks, and high fault rates. Non-deterministic route paths, changing link availability, and dynamic participation of nodes complicate network performance management. These dynamics necessitate routes with adaptive capabilities to respond to environmental uncertainties and maintain low energy consumption and high resilience.

WSN routing protocols are traditionally classified by network structure (flat, hierarchical, location-based), but whatever their structure, the protocols have to deal with fundamental issues like Quality of Service (QoS), scalability, delay, power efficiency, connectivity, and fault tolerance. Network topology inconsistency caused by the addition or removal of motes or transient link failures makes path selection even more difficult. Thus, the routing technique has to account for a broad range of parameters like power management, MAC protocols, traffic load, and channel allocation to develop implementable solutions.

With these theoretical and interdependent variables in mind, heuristic and metaheuristic algorithms have been explored as potential methods for designing a WSN routing protocol. This study introduces an energy-efficient framework developed on a multi-objective problem optimized by the ECSA. The framework is focused on optimal CH selection and energy-efficient data routing to achieve maximum network lifespan while fulfilling performance requirements.

Our proposed scheme uses a layered clustering paradigm. CHs collect intra-cluster data and transfer it to the BS by energy-

efficient multi-hop routes. Our ECSA dynamically selects CHs using residual energy, node centrality, and communication distance parameters. The approach optimizes competing objectives like energy consumption and throughput maximization, thus constructing a resilient and dynamic routing protocol designed to work in WSN scenarios.

A. Basic Crow Search Algorithm

CSA, proposed by Askarzadeh [24], is a population-based metaheuristic inspired by the natural foraging and memory behavior of crows. These intelligent birds can follow others to steal hidden food while relocating their food caches if they sense being watched. This dual behavior of tracking and evasion is simulated within CSA to navigate the search space intelligently.

In CSA, each solution is represented as a virtual crow, and a population of N such crows explores the search space. Each crow i is characterized by a position vector in a D-dimensional space, denoted by Eq. (10).

$$x_i^t = [x_{i,1}^t, x_{i,2}^t, \dots, x_{i,D}^t] \tag{10}$$

Each crow's initial position also serves as its memory since it has no experience. The memory vector of crow i at iteration t is expressed as shown in Eq. (11).

$$m_i^t = [m_{i,1}^t, m_{i,2}^t, \dots, m_{i,D}^t] \tag{11}$$

The crows update their positions by attempting to follow a randomly selected peer. If the selected crow is unaware of being followed, the follower moves toward the peer's memory. Otherwise, it jumps to a random position. This movement rule is calculated according to Eq. (12).

$$x_i^{t+1} = \begin{cases} x_i^t + r_1 . FL . (m_j^t - x_i^t), & if\ r_2 \geq AP \\ random\ position\ , & otherwise \end{cases} \tag{12}$$

where, r_1 and r_2 are random values uniformly distributed in [0, 1], FL is the flight length, m_j^t is the memory of randomly selected crow j, and AP is the awareness probability governing evasive behavior. The schematic of this condition and the impact of flight length on searchability are shown in Fig. 2.

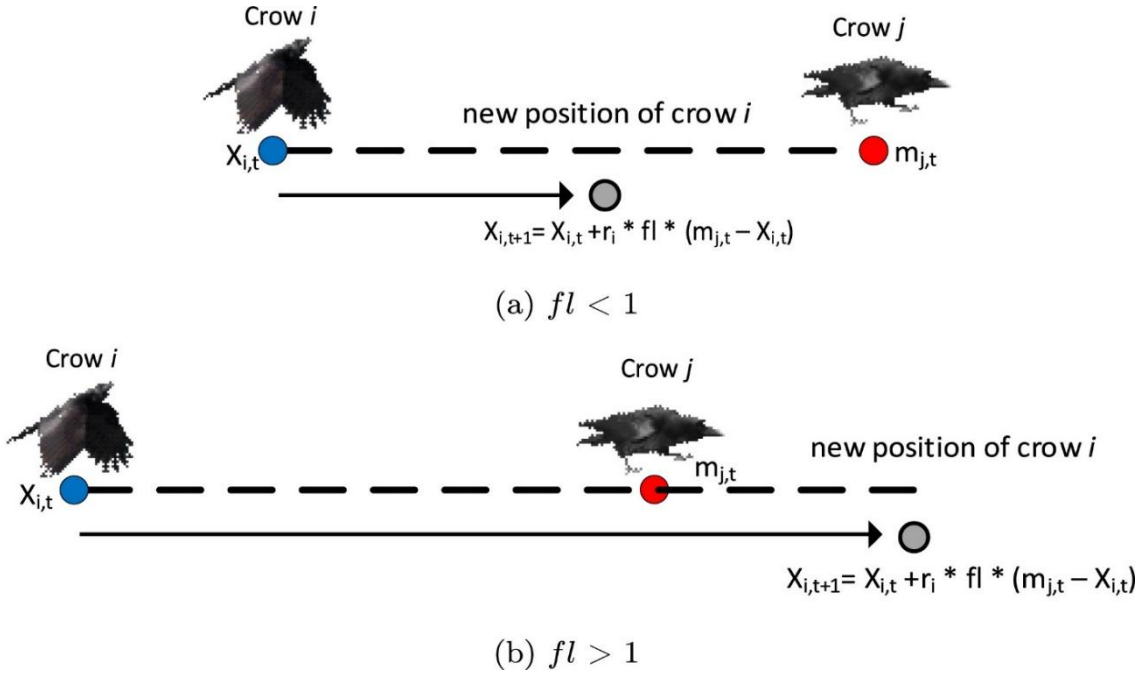


Fig. 2. Illustration of the pursuit mechanism in CSA.

After arriving at new positions, the memory of each crow is updated based on the quality of the new solution. If the new position offers better fitness than the previous memory, it replaces the old one. This memory update mechanism is governed by Eq. (13).

$$m_i^{t+1} = \begin{cases} x_i^{t+1}, & if\ f(x_i^{t+1}) < f(m_i^t) \\ m_i^t, & otherwise \end{cases} \tag{13}$$

where, $f(.)$ indicates the objective (fitness) function.

B. Enhanced Crow Search Algorithm

While the original CSA has high exploration capabilities, it has drawbacks like local optima stagnation, lack of diversity, and early-stage convergence. These problems are caused primarily by fixed parameters like Awareness Probability (AP) and Flight Length (FL), using random guidance, and not prioritizing the optimal use of population memory. To overcome all these drawbacks, ECSA incorporates a range of strategies, such as adaptive selection through a tournament and a modified evasion scheme using a random component.

In the basic CSA, each crow i selects another crow j at random to follow toward its memorized position, which could

lead to following poorly performing solutions. To improve this, the ECSA framework replaces random selection with a fitness-guided adaptive tournament mechanism. At each iteration t, a crow selects the best guide from a randomly chosen subset of K crows.

To ensure a balanced tradeoff between exploration and exploitation, the value of K is varied linearly throughout the optimization process. It is computed using a self-adaptive formula denoted by Eq. (14).

$$K = round\left(K_{min} + t.\left(\frac{K_{max} - K_{min}}{t_{max}}\right)\right) \quad (14)$$

Where K_{min} and K_{max} are the minimum and maximum tournament sizes, t is the current iteration, and t_{max} is the total number of iterations. This allows the algorithm to perform more exploratory behavior in the early stages (small K) and intensify the search later (large K).

To further combat stagnation, ECSA introduces an evasion operator derived from the Harris Hawks Optimization (HHO) method, which improves population diversity when the awareness probability condition is triggered. The new position of crow i in this phase is calculated according to Eq. (15).

$$X_{i,t+1} = \left(X_{best,t} - X_{avg,t}\right) - r_1 \cdot \left(LB + r_2 \cdot (UB - LB)\right) \quad (15)$$

where, $X_{best,t}$ is the best-known solution at iteration t, $X_{avg,t}$ is the mean position of all individuals in the current generation, LB and UB are the lower and upper bounds of the search space, and $r_1, r_2 \in [0,1]$ are uniformly distributed random numbers. The mean position is calculated as shown in Eq. (16).

$$X_{avg,t} = \frac{1}{N}\sum_{i=1}^{N} X_{i,t} \quad (16)$$

This strategy ensures that evasion is not purely random but guided by the global and average population dynamics.

ECSA begins by initializing N crows in a D-dimensional search space. Each crow's initial position is generated using a uniformly distributed random vector within bounds $\overrightarrow{X_L}$ and $\overrightarrow{X_U}$, as denoted by Eq. (17).

$$\overrightarrow{X_J} = \overrightarrow{X_L} + r \cdot \left(\overrightarrow{X_U} - \overrightarrow{X_L}\right) \quad (17)$$

To further enhance search efficiency and diversity preservation, ECSA incorporates an island model. The population is partitioned into s subgroups (islands), each running its local version of the ECSA. These islands evolve independently but periodically share information through a migration process governed by migration frequency M_f, migration rate M_r, topology (e.g. random ring), and policy (e.g. best–worst exchange) parameters. Each migration phase is triggered every M_f iterations, allowing selected individuals from one island to be exchanged with neighboring islands to prevent premature convergence. The pseudocode and flowchart of the proposed ECSA are presented in Algorithm 1 and Fig. 3, respectively.

Algorithm 1 Pseudocode of ECSA

Inputs:

Problem-specific settings: number of agents N, dimensionality d, upper bound t_{max}

Adaptive parameters: minimum and maximum tournament size (K_{min}, K_{max}), flight length FL, and awareness probability AP

Outputs:

Best solution vector X_{best} and its corresponding fitness score

Initialize all crows' positions X_i randomly across the solution space, for $i = 1\ to\ N$

Assign each crow's initial memory $M_i = X_i$

Evaluate the fitness value of each crow using the objective function $f(\cdot)$

Determine the initial global best solution X_{best}

Repeat for iteration $t = 1\ to\ t_{max}$:

Adjust tournament size K dynamically based on Eq. 14

Compute the population's average position using Eq. 16

For each crow $i \in \{1, \ldots, N\}$:

Select a candidate crow j to follow via tournament selection

Generate a random number $r_j \in [0,1]$

If $r_j \geq AP$:

Update crow i's position using the pursuit rule (Eq. 12)

Else:

Update crow i's position using guided evasion (Eq. 15)

Check feasibility of the new position

Evaluate the updated positions X_i^{t+1} and compute their fitness

If $f\left(X_i^{t+1}\right) < f(M_i)$, then update the memory $M_i = X_i^{t+1}$

Update X_{best} if any crow achieves a superior fitness

Until stopping criteria ($t = t_{max}$) is met

Return the best-obtained solution X_{best}

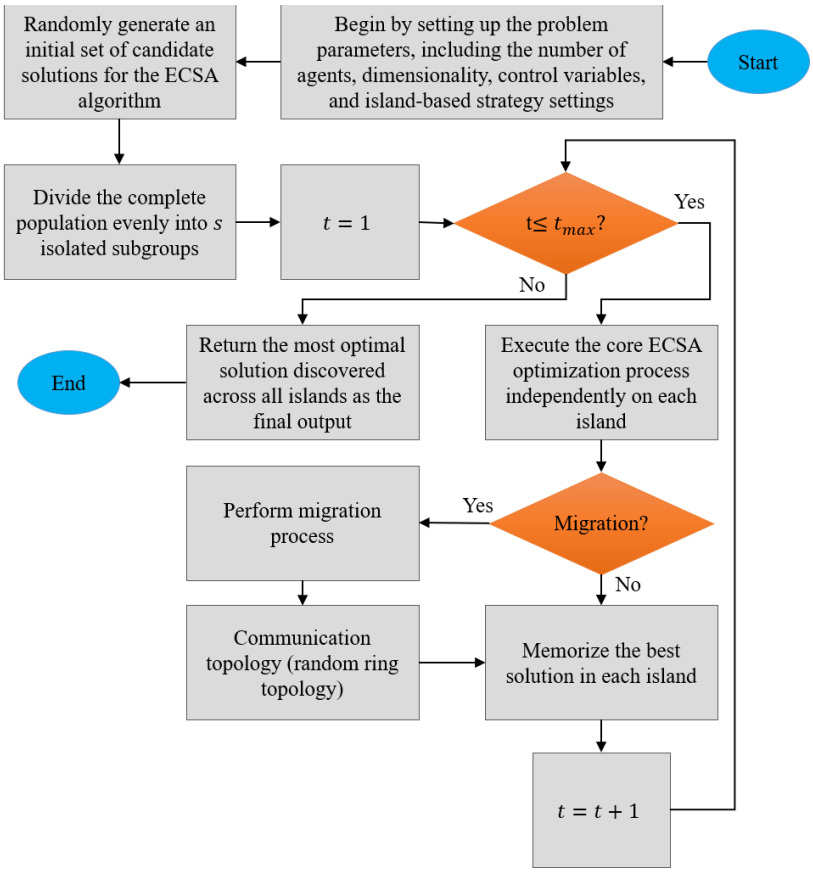


Fig. 3. Flowchart of CSA.

V. Results

A comprehensive simulation in MATLAB 2020a was performed to assess the effectiveness of the proposed ECSA for energy-saving routing within WSNs. Performance was measured using a range of vital metrics, including convergence behavior, path loss, PDR, throughput, and energy usage. For extensive investigation, ECSA was compared with four metaheuristic algorithms: CSA, Grey Wolf Optimizer (GWO) [25], Black Widow Optimization (BWO) [26], and Whale Optimization Algorithm (WOA) [27]. These algorithms were chosen because of their suitability for WSN routing and clustering purposes. Simulations were performed on three distinct network configurations with varying sensor nodes deployed: Experiment 1 with 50 nodes, Experiment 2 with 100 nodes, and Experiment 3 with 150 nodes. Table II lists simulation parameters.

As illustrated in Fig. 4, ECSA consistently achieves a high convergence score across all node arrangement scenarios. In particular, in Scenario 1, ECSA surpasses the comparative methods by 55.2%, 51.4%, 48.4%, and 41.3%. This indicates the algorithm's stability in reaching optimal or near-optimal solutions rapidly. The increased convergence comes from ECSA's adaptive exploration-exploitation tradeoff to avoid premature convergence inherent in conventional methods.

TABLE II. Simulation Setup Parameters Applied in Evaluating the Proposed Routing Framework for WSN

Parameter	Configured value
Maximum simulation rounds	2000
Initial node energy	0.3 J
Energy dissipation for signal transmission	$50 \times 10^{-9} J$
Energy dissipation for signal reception	$50 \times 10^{-9} J$
Power amplifier (free-space model)	$10 \times 10^{-12} J$
Power amplifier (multi-path model)	$0.0013 \times 10^{-12} J$
Energy cost for data aggregation	$5 \times 10^{-9} J$
Energy consumption in idle state	0.05 W
Node displacement rate	0.05 meters/min
Channel model	Physical/wireless physical
Communication radius	40 meters
Underlying routing baseline	LEACH protocol
Simulation area size	$100\ m \times 100\ m$
Sensor node counts across scenarios	50, 100, 150
Control message size	80 bits
Data packet size	512 bytes
Cluster head election likelihood	0.1%
Optimization iterations per run	10
Sub-populations for the metaheuristic algorithm	10

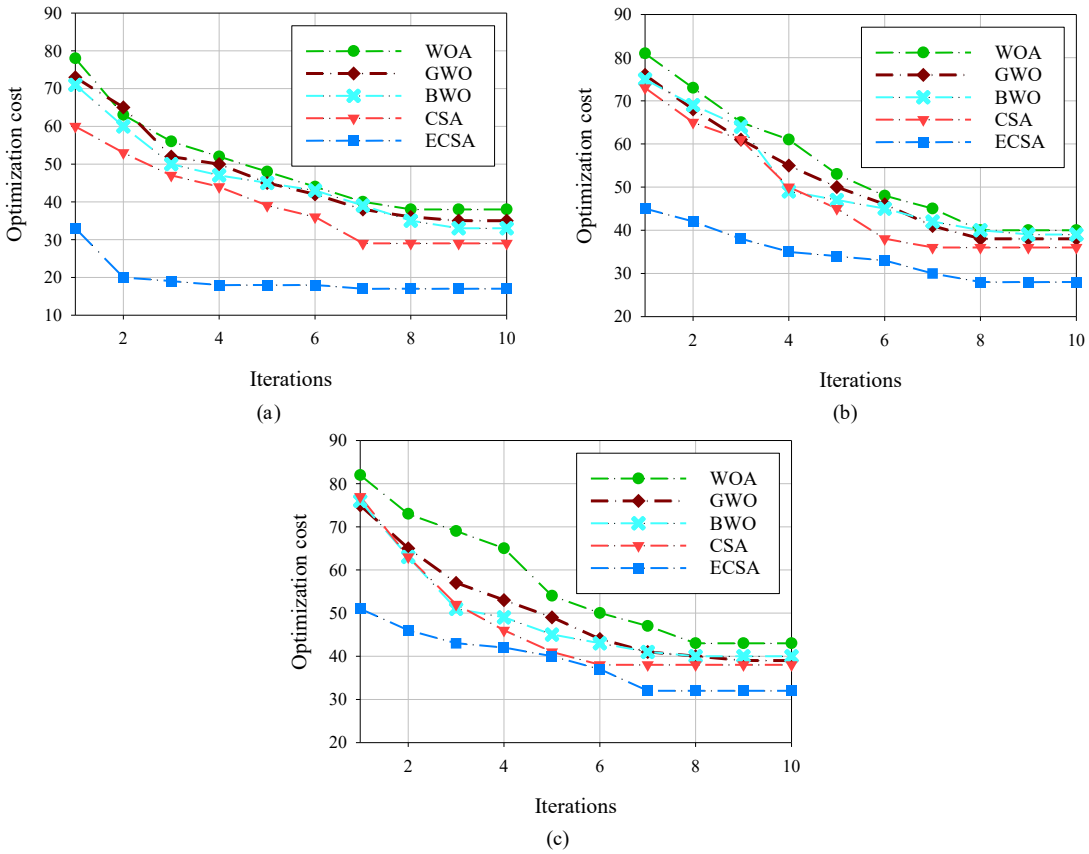


Fig. 4. Cost function convergence of algorithms across three WSN scenarios: (a) 50 sensor nodes, (b) 100 sensor nodes, (c) 150 sensor nodes.

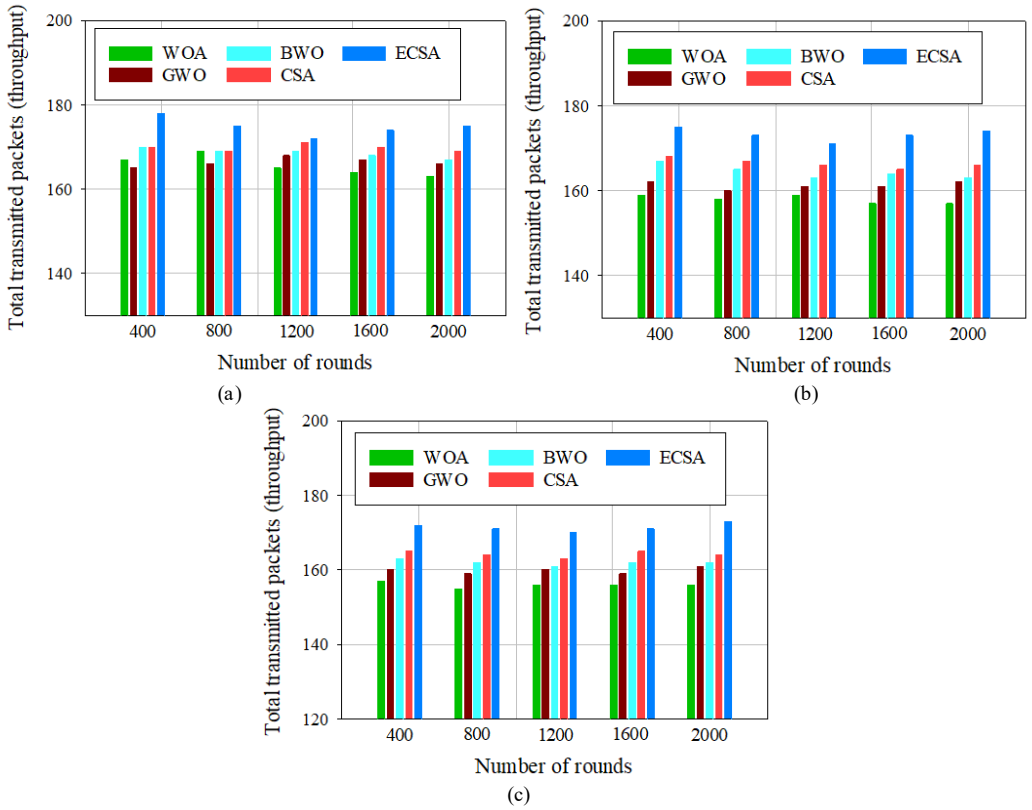


Fig. 5. Throughput performance of algorithms across different rounds for (a) 50 nodes, (b) 100 nodes, and (c) 150 nodes.

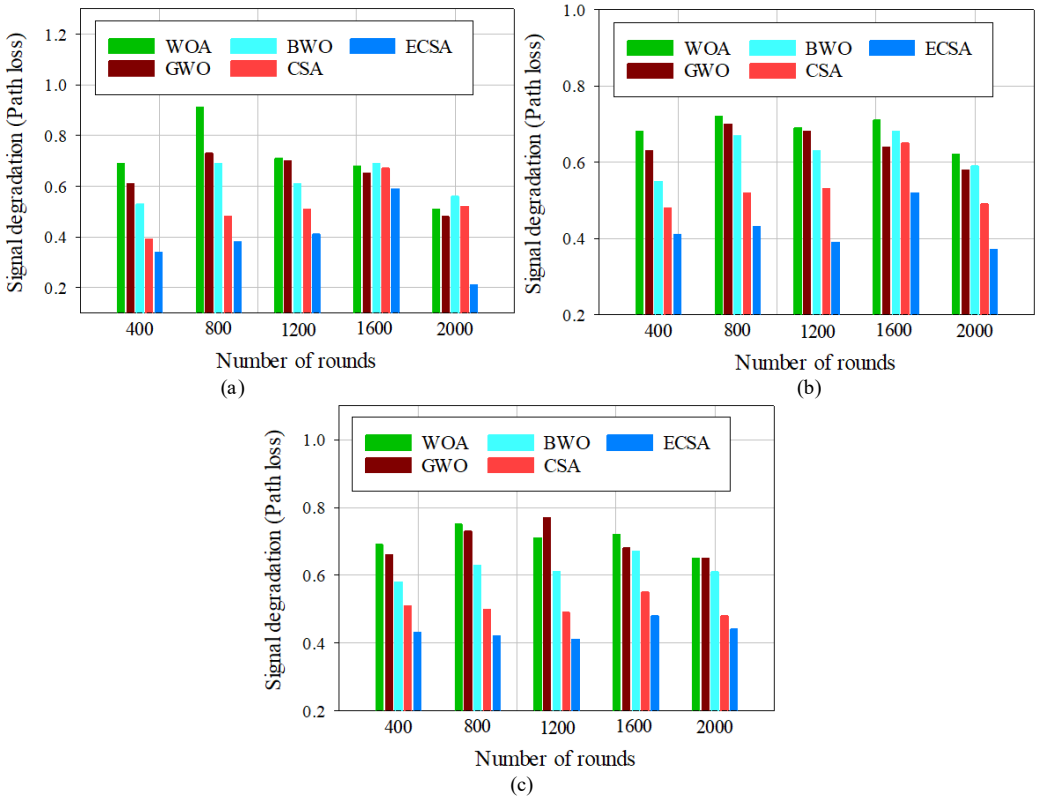


Fig. 6. Path loss performance of algorithms across different rounds for (a) 50 nodes, (b) 100 nodes, and (c) 150 nodes.

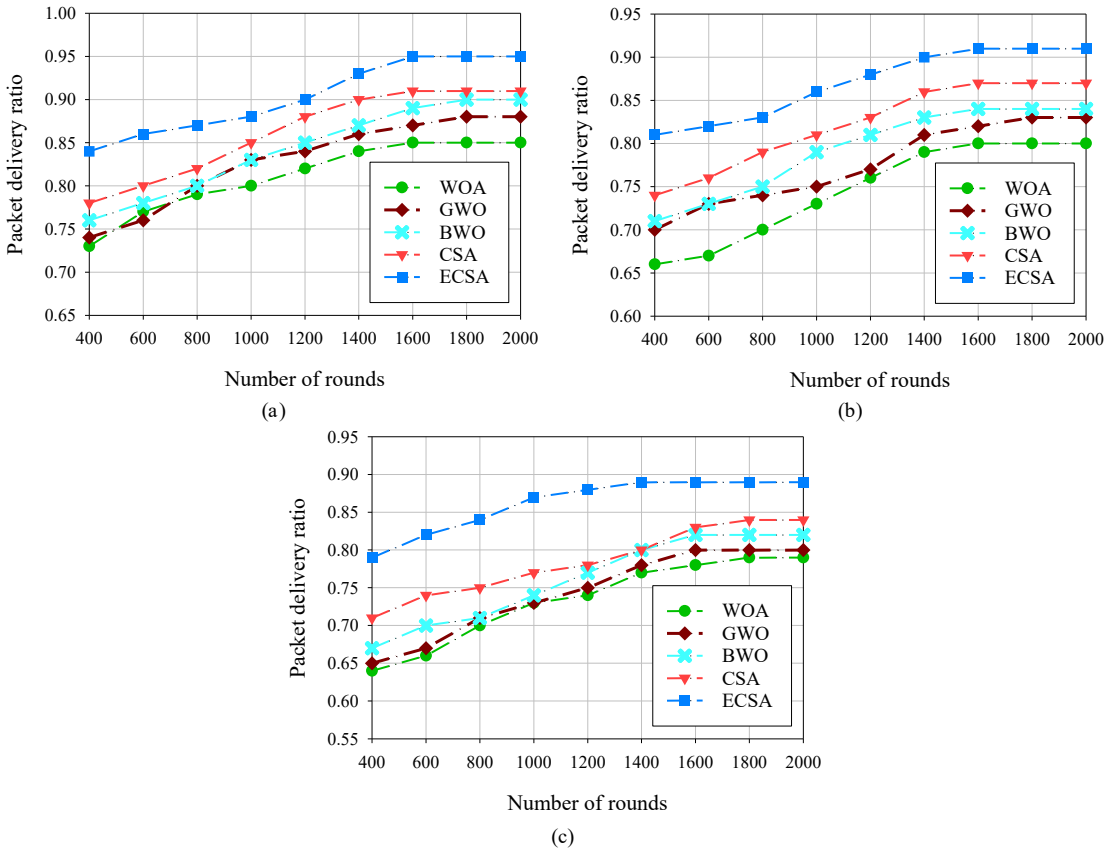


Fig. 7. Packet delivery ratio performance of algorithms across different rounds for (a) 50 nodes, (b) 100 nodes, and (c) 150 nodes.

The performance of ECSA in improving network throughput was confirmed by evaluating the volume of data delivered successfully against time. Fig. 5 shows that ECSA outpaces heuristic algorithms consistently by achieving superior throughput in all cases. Better throughput is the direct consequence of smart CH election, less path loss, and energy-aware transmission scheduling. This proves ECSA's ability to sustain data-intensive WSN applications with improved transmission efficiency.

The path loss was compared under three node-density configurations as a representative factor of signal quality and energy consumption. As shown in Fig. 6, ECSA outperforms in reducing path loss until the 2000th round. Furthermore, in the first and second scenarios, the path loss is temporarily higher than in CSA at the advanced stages, and it might indicate a necessity to refine it for longer-duration deployments in the future. The ECSA shows fewer path loss values and validates the proposed routing model's credibility and robustness.

The data delivery reliability of the protocol was checked using PDR, which is presented in Fig. 7. ECSA has a consistently low rate of losing PDR compared to other routing protocols in all scenarios. The reduced rate of losing PDR results from the algorithm's stability and efficient selection of the CH, reducing the rate of dropping and discarding packets in communication. Therefore, ECSA provides superior delivery reliability in critical WSN applications.

The protocol's sustainability was analyzed by measuring the active node count per simulation round. As indicated by Fig. 8, ECSA retained a significantly higher number of alive nodes across the three cases than baseline strategies. In scenario 2, ECSA offers superior node survival across transmission rounds. This demonstrates its energy-conservation properties and suggests fewer nodes reach their energy boundaries earlier. Superior node survivability confirms the efficiency of ECSA in balancing energy load distributions when choosing CHs.

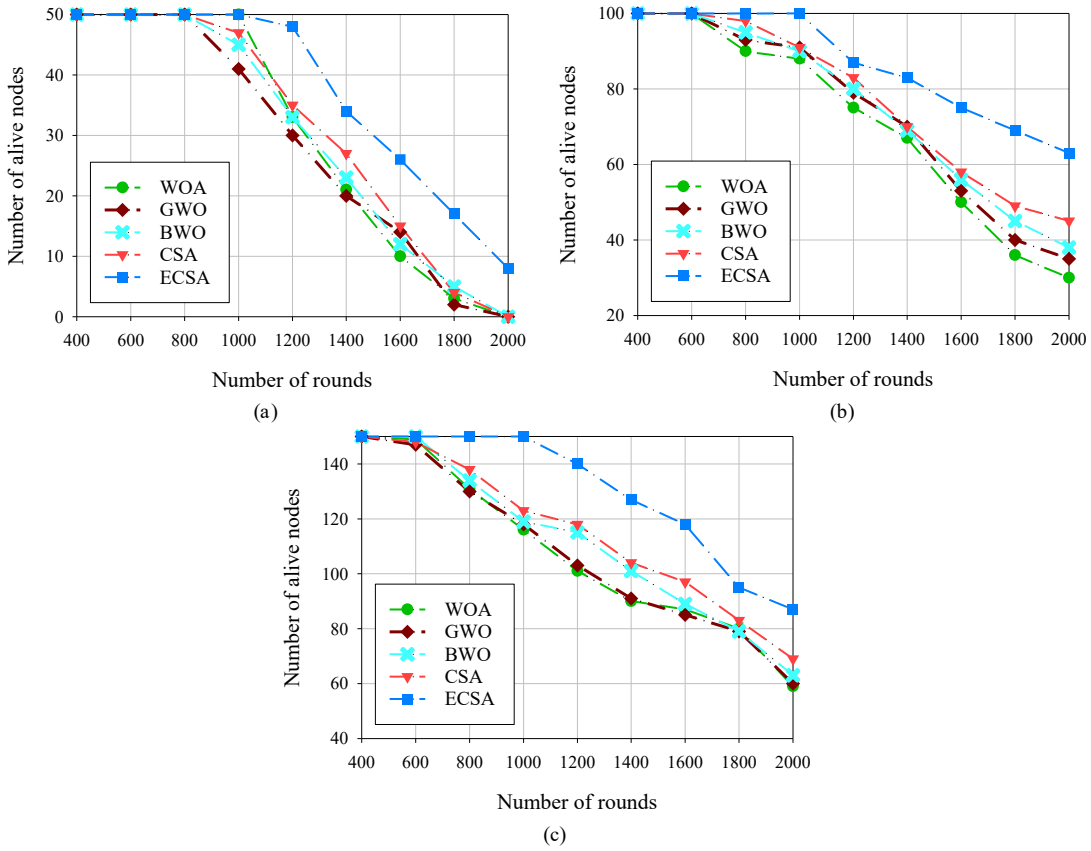


Fig. 8. Number of alive nodes of algorithms across different rounds for (a) 50 nodes, (b) 100 nodes, and (c) 150 nodes.

VI. Discussion

The superior simulation results of ECSA on all evaluation parameters can directly be attributed to intrinsic advancements in CSA. One key aspect involves adding an adaptive tournament selection policy to facilitate selection for updated positions. Compared to the initial CSA, in which randomly selected peers guide crows, ECSA allows crows to be guided to the fittest peers from a dynamically sized pool of candidates. Indeed, as demonstrated in the convergence results, the technique resulted in quicker convergence and more robust solution quality, particularly when network densities are high. Moreover, the linear increase in the tournament size per iteration permitted the algorithm to expand its search at the beginning and intensify exploitation towards the latter phase, which lent it to minimal energy consumption and maximum node survival in all test cases.

In addition, the advanced evasion process utilized by ECSA contributed significantly to optimizing route performance. In contrast to the standard CSA that triggers random relocation on

evasion, ECSA incorporates a focused evasion process by using global best and average location information for the population. This ensures informed and active exploratory behavior even when AP is high. Consequently, ECSA resulted in lower path loss and higher PDR, particularly at high node density and extended round simulation. Improved throughput performance further confirms that the advancements in pursuit (guide selection) and evasion (movement policy) acted synergistically to produce superior route decisions, energy balancing, and overall robust protocol performance in WSN scenarios.

Though the proposed ECSA-based routing protocol provides superior energy efficiency and robustness over existing methods, the study has some drawbacks. First, it was conducted within simulated networks with pre-assumed network models that do not capture fundamental WSN dynamics. Furthermore, although the cooperative island model provides superior population diversity, computational complexity increases with the size of the network, such that large-scale networks may lack scalability in very dense sensor environments. Finally, security-related issues, such as resistance to malicious nodes, have not been specifically addressed. These drawbacks present potential areas for future work, such as hardware-based verification, optimization for scalability, and the addition of security-aware mechanisms.

VII. Conclusion

The paper introduced ECSA for energy-efficient routing in WSNs by hybridizing a cooperating island model with adaptive tournament selection. From a theoretical standpoint, the paper first applied a structured population model in CSA for CH selection to address two essential problems: premature convergence and limited population diversity. Apart from WSNs, the model provides a generic framework for integrating swarm intelligence strategies with hierarchical clustering mechanisms. An experimental study on networks containing 100 nodes confirmed that ECSA outperformed CSA, BWO, GWO, and WOA. In detail, ECSA achieved 22–30% lower optimization cost, 4.8–10.8% higher throughput, 24.4–40.3% lower path loss, 4.5–13.7% higher PDR, and 40.1–109.1% more alive nodes. These results confirm the robustness, scalability, and energy-saving capabilities of the approach if applied at a large scale in WSN.

In addition, the study also has its limitations. The analysis was conducted in a simulated environment without external dynamics, such as hardware variation, environmental interference, or malicious nodes. Furthermore, the cooperative island model entails additional computational overhead that may hinder scalability in ultra-dense networks. The future study will therefore focus on 1) protocol verification over hardware testbeds, 2) lowering computational complexity for optimization of scalability, as well as 3) extending the model for the addition of security-aware routing.

Funding

This work was supported by project of Henan Province scientific and technological research. (No. 252102210202).

References

[1] M. N. Mowla, N. Mowla, A. S. Shah, K. M. Rabie, and T. Shongwe, "Internet of Things and wireless sensor networks for smart agriculture applications: A survey," IEEe Access, vol. 11, pp. 145813–145852, 2023.

[2] S. R. Jondhale, R. Maheswar, J. Lloret, S. R. Jondhale, R. Maheswar, and J. Lloret, "Fundamentals of wireless sensor networks," Received Signal Strength Based Target Localization and Tracking Using Wireless Sensor Networks, pp. 1–19, 2022.

[3] S. U. Khan, Z. U. Khan, M. Alkhowaiter, J. Khan, and S. Ullah, "Energy-efficient routing protocols for UWSNs: A comprehensive review of taxonomy, challenges, opportunities, future research directions, and machine learning perspectives," Journal of King Saud University-Computer and Information Sciences, p. 102128, 2024.

[4] B. Pourghebleh, N. Hekmati, Z. Davoudnia, and M. Sadeghi, "A roadmap towards energy‐efficient data fusion methods in the Internet of Things," Concurrency and Computation: Practice and Experience, vol. 34, no. 15, p. e6959, 2022.

[5] P. M. Kumar, T. Shahwar, and C. Gokulnath, "Improved sensor localization with intelligent trust model in heterogeneous wireless sensor network in Internet of Things (IoT) environment," Sustainable Computing: Informatics and Systems, vol. 46, p. 101122, 2025.

[6] V. Rajaram et al., "Enriched energy optimized LEACH protocol for efficient data transmission in wireless sensor network," Wireless Networks, vol. 31, no. 1, pp. 825–840, 2025.

[7] W. B. Nedham and A. K. M. Al-Qurabat, "A comprehensive review of clustering approaches for energy efficiency in wireless sensor networks," International Journal of Computer Applications in Technology, vol. 72, no. 2, pp. 139–160, 2023.

[8] L. Zhang, J. Qi, and H. Wu, "A novel data aggregation method for underwater wireless sensor networks using ant colony optimization algorithm," International Journal of Advanced Computer Science and Applications, vol. 14, no. 4, 2023.

[9] W. Fang, W. Zhang, W. Yang, Z. Li, W. Gao, and Y. Yang, "Trust management-based and energy efficient hierarchical routing protocol in wireless sensor networks," Digital Communications and Networks, vol. 7, no. 4, pp. 470–478, 2021.

[10] Y. Raghavendra and U. Mahadevaswamy, "Energy efficient intra cluster gateway optimal placement in wireless sensor network," Wireless Personal Communications, vol. 119, no. 2, pp. 1009–1028, 2021.

[11] S. Regilan and L. Hema, "Optimizing energy efficiency and routing in wireless sensor networks through genetic algorithm-based cluster head selection in a grid-based topology," Journal of High Speed Networks, vol. 30, no. 4, pp. 569–582, 2024.

[12] A. S. Nandan, S. Singh, R. Kumar, and N. Kumar, "An optimized genetic algorithm for cluster head election based on movable sinks and adjustable sensing ranges in IoT-based HWSNs," IEEE Internet of Things Journal, vol. 9, no. 7, pp. 5027–5039, 2021.

[13] H. Hu, X. Fan, C. Wang, T. Wang, and Y. Deng, "Particle swarm optimization and fuzzy logic based clustering and routing protocol to enhance lifetime for wireless sensor networks," Cluster Computing, vol. 27, no. 7, pp. 9715–9734, 2024.

[14] M. Mann and R. Singh, "Optimized stochastic particle swarm optimization integrated ranked backoff medium access control for optimal route selection in wireless sensor networks," Peer-to-Peer Networking and Applications, vol. 18, no. 1, p. 30, 2025.

[15] M. K. Sahu and S. Patil, "Enhanced double cluster head selection using ant-colony optimization for energy-efficient routing in wireless sensor network," SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, vol. 13, no. 01, pp. 35–41, 2021.

[16] V. Hayyolalam, B. Pourghebleh, and A. A. Pourhaji Kazem, "Trust management of services (TMoS): investigating the current mechanisms," Transactions on Emerging Telecommunications Technologies, vol. 31, no. 10, p. e4063, 2020, doi: https://doi.org/10.1002/ett.4063.

[17] S. R. Nabavi, V. Ostovari Moghadam, M. Yahyaei Feriz Hendi, and A. Ghasemi, "Optimal selection of the cluster head in wireless sensor networks by combining the multiobjective genetic algorithm and the gravitational search algorithm," Journal of Sensors, vol. 2021, no. 1, p. 2292580, 2021.

[18] P. Kathiroli and K. Selvadurai, "Energy efficient cluster head selection using improved Sparrow Search Algorithm in Wireless Sensor Networks," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 10, pp. 8564–8575, 2022.

[19] S. Janakiraman, "Improved bat optimization algorithm and enhanced artificial bee colony‐based cluster routing scheme for extending network lifetime in wireless sensor networks," International journal of communication systems, vol. 36, no. 5, p. e5428, 2023.

[20] T. Luo, J. Xie, B. Zhang, Y. Zhang, C. Li, and J. Zhou, "An improved levy chaotic particle swarm optimization algorithm for energy-efficient cluster routing scheme in industrial wireless sensor networks," Expert Systems with Applications, vol. 241, p. 122780, 2024.

[21] A. Srivastava and P. K. Mishra, "IDA: Improved dragonfly algorithm for load balanced cluster heads selection in wireless sensor networks," Peer-to-Peer Networking and Applications, vol. 17, no. 4, pp. 2315–2333, 2024.

[22] H. Alsuwat and E. Alsuwat, "Energy-aware and efficient cluster head selection and routing in wireless sensor networks using improved artificial bee Colony algorithm," Peer-to-Peer Networking and Applications, vol. 18, no. 2, pp. 1–24, 2025, doi: https://doi.org/10.1007/s12083-024-01810-y.

[23] P. Poonguzhali, P. Geetha, and R. Vidhya, "Multi-objective cluster head based energy efficient routing using THDCNN with hybrid capuchin search and woodpecker mating algorithm in WSN," Wireless Networks, pp. 1–17, 2025.

[24] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm," Computers & structures, vol. 169, pp. 1–12, 2016.

[25] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in engineering software, vol. 69, pp. 46–61, 2014.

[26] V. Hayyolalam and A. A. P. Kazem, "Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems," Engineering Applications of Artificial Intelligence, vol. 87, p. 103249, 2020.

[27] S. Mirjalili and A. Lewis, "The whale optimization algorithm," Advances in engineering software, vol. 95, pp. 51–67, 2016, doi: https://doi.org/10.1016/j.advengsoft.2016.01.008.

From Review to Practice: A Comparative Study and Decision-Support Framework for Sentiment Classification Models

Kamal Walji[1], Allae Erraissi[2], Abdelali ZAKRANI[3], Mouad Banane[4]
Laboratory of Artificial Intelligence & Complex Systems Engineering, Hassan II University,
ENSAM and Casablanca, Morocco[1, 3, 4]
Chouaib Doukkali University, El Jadida, Morocco[2]

***Abstract*—Sentiment classification is a core task in natural language processing (NLP), enabling automated interpretation of opinionated text across domains, such as social media, e-commerce, and healthcare. While numerous models have been proposed—from classical machine learning algorithms to deep neural networks and transformer architectures—their adoption is often hindered by trade-offs in performance, interpretability, and computational cost. This paper presents a threefold contribution: 1) a structured review of over 30 peer-reviewed studies that compare sentiment classifiers across five analytical dimensions—accuracy, robustness, interpretability, efficiency, and context adaptability; 2) a lightweight empirical benchmark on the IMDb dataset, evaluating Naïve Bayes, linear SVM, and LSTM; and 3) a practitioner-oriented decision-support framework comprising a model selection flowchart and recommendation matrix. The experimental results show that SVM achieved the highest F1-score (0.8329), while Naïve Bayes provided strong performance with minimal training time, and LSTM underperformed under constrained conditions. We further highlight persistent challenges in benchmarking consistency, model explainability, and cross-lingual adaptability. The paper concludes with actionable future directions, including hybrid architectures, low-resource deployment strategies, and inclusive NLP systems for diverse user populations. To our knowledge, this is the first study that unifies systematic review, empirical validation, and practical decision tools in the field of sentiment classification.**

Keywords*—Sentiment analysis; text classification; machine learning; deep learning; transformer models; BERT; LSTM; random forest; hybrid approaches; model evaluation; interpretability; natural language processing

I. Introduction

The exponential growth of user-generated content on digital platforms has amplified the role of sentiment analysis (SA) in natural language processing (NLP). From monitoring public sentiment on social media to enhancing customer experience in e-commerce and health feedback systems, the ability to computationally interpret opinion-laden text has become crucial. The evolving landscape of machine learning (ML) and deep learning (DL) has significantly advanced sentiment classification, yet persistent challenges hinder broader adoption and practical deployment.

Key challenges include the ability to handle heterogeneous and multilingual datasets, mitigate the effects of class imbalance, and effectively model contextual dependencies. Classical ML models, while efficient and interpretable, often struggle with generalization in dynamic environments. Conversely, advanced neural architectures such as long short-term memory networks (LSTM), convolutional neural networks (CNN), and transformers demonstrate superior performance in complex settings but require extensive computational resources and lack interpretability.

Recent years (2024–2025) have witnessed a surge in hybrid models combining traditional and deep learning techniques. These models seek to balance performance, adaptability, and computational efficiency, and are increasingly relevant for real-world applications demanding both accuracy and scalability.

This paper contributes a comprehensive, multi-dimensional review of sentiment classification models. It uniquely focuses on hybrid architectures developed in the latest period and proposes a comparative framework for evaluating models across five dimensions: accuracy, robustness, interpretability, efficiency, and context adaptability.

The study addresses the following research questions:

- RQ1: How do hybrid and standalone deep learning models perform compared to classical algorithms in real-world sentiment classification tasks?
- RQ2: What are the practical trade-offs in deploying sentiment classification models across diverse domains?
- RQ3: What guidelines can be established for context-aware model selection in sentiment classification?

By addressing these questions, this study aims to inform both researchers and practitioners on the evolving capabilities of sentiment classification systems and offer guidance for selecting models that align with specific deployment requirements.

This paper makes three key contributions toward advancing sentiment classification research and practice:

1) A structured, paradigm-based literature review of over 30 peer-reviewed studies, comparing classical machine learning, deep learning, and transformer-based models across five analytical dimensions: accuracy, robustness, interpretability, computational efficiency, and context adaptability.

2) An independent, lightweight benchmark conducted on the IMDb dataset, empirically comparing three representative models—Naïve Bayes, linear SVM, and LSTM—under constrained training conditions to validate practical trade-offs observed in the literature.

3) A practitioner-oriented decision-support framework, consisting of a flowchart and recommendation matrix, designed to guide model selection based on resource constraints, domain specificity, and sequence modeling needs.

This study unifies a survey, empirical benchmarking, and practical model selection guidance into a coherent framework for sentiment analysis. By bridging academic research with real-world deployment considerations, this work provides actionable insights for both researchers and practitioners.

The remainder of this paper is structured as follows: Section II reviews prior research on sentiment classification models, covering classical machine learning, deep learning, ensemble, and transformer-based approaches. Section III outlines the study methodology, including literature selection criteria, inclusion conditions, and the comparative evaluation framework. Section IV presents the algorithmic landscape, analyzing strengths, weaknesses, and typical use cases of key sentiment classification algorithms. Section V reports and discusses the results from both the literature synthesis and the empirical benchmark on the IMDb dataset, along with performance comparisons and confusion matrix insights. Section VI concludes the study by summarizing key findings, discussing future directions, and reinforcing the practical utility of the proposed decision-support framework.

II. Related Works

A. A Critical Review of Sentiment Analysis Methods

This review is grounded in a structured selection of peer-reviewed publications relevant to the field of sentiment analysis. This review draws upon peer-reviewed publications published between 2016 and 2025 from recognized venues, including studies indexed in IEEE, ACM, and open-access platforms. Inclusion criteria were based on relevance to sentiment classification using classical, deep learning, or transformer-based models. Studies had to report empirical results using public datasets and standardized metrics such as accuracy, F1-score, or precision. After screening and eliminating duplicates, a representative subset of the most relevant and influential works was analyzed to synthesize key findings, identify limitations, and highlight research gaps addressed in this study.

Sentiment analysis has evolved significantly, transitioning from classical statistical models to deep learning and transformer-based architectures. Each paradigm offers distinct advantages and presents specific challenges, particularly in terms of contextual understanding, scalability, and robustness. This section offers a critical synthesis of these developments, integrating insights from a representative set of peer-reviewed studies published between 2016 and 2025 across IEEE, ACM, and arXiv. The organization follows a paradigm-based structure, focusing on unresolved gaps that inform the methodological choices of our study.

B. Classical ML Approaches

Foundations and Limitations Naïve Bayes, Logistic Regression, and SVM remain favored for their simplicity and speed. However, their dependence on feature engineering and inability to capture semantic subtleties limit their utility in dynamic textual environments. A broad comparative study by [1] evaluates classical supervised algorithms—including Naive Bayes, Decision Trees, Random Forest, KNN, and SVM—across twelve criteria such as efficiency, robustness, and interpretability. Their findings highlight that the optimal algorithm depends on task-specific requirements, available resources, and data characteristics, confirming the continued relevance of classical ML in constrained environments.

C. Ensemble Methods

Combining Simplicity and Strength Random Forests and similar ensembles show resilience against noise and imbalance. Yet, their black-box nature and hyperparameter sensitivity hinder interpretability. A complementary comparative analysis by [2] reinforces these points, particularly noting the robustness of Random Forest and the context sensitivity of LSTM, and offers practical insights into real-time deployments.

D. Deep Learning Models

Contextual Learning with Neural Networks Deep learning has significantly transformed sentiment analysis by enabling end-to-end training without manual feature engineering. Convolutional Neural Networks (CNNs), first adopted by [3] for text classification, have demonstrated strong performance in capturing local syntactic and semantic features, particularly in short texts and tweets. Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models, are widely used for learning sequential dependencies in sentiment-laden texts [4] LSTMs are particularly adept at handling long-range dependencies and have become standard in many SA pipelines. However, their training complexity and sensitivity to vanishing gradients remain limitations. More recent efforts, such as bidirectional LSTM (BiLSTM) and attention-augmented RNNs [5], attempt to overcome these challenges by capturing both forward and backward contextual information.

E. Transformer Architectures

High Accuracy at Computational Cost Transformer-based architectures have redefined state-of-the-art in sentiment analysis, particularly since the introduction of Bidirectional Encoder Representations from Transformers (BERT) by [6]. Unlike RNNs, transformers use self-attention mechanisms to capture global dependencies and contextual relationships without relying on sequential processing. Variants such as RoBERTa [5], ALBERT [7], and DistilBERT [8] optimize transformer models for speed and efficiency while preserving performance. These models have consistently achieved top scores on benchmark datasets like SST-2, Yelp, and Amazon reviews. Despite their accuracy, the computational demands of transformer models—both in training and inference—pose significant barriers to deployment in low-resource settings. Moreover, their interpretability remains a challenge, though recent methods (e.g. attention visualization, SHAP explanations) attempt to address this.

F. The Emerging Middle Ground

Hybrid models are increasingly recognized as a practical compromise between accuracy and computational efficiency in sentiment classification. Architectures that combine Convolutional Neural Networks (CNNs), Bidirectional LSTMs (BiLSTMs), and attention mechanisms are particularly effective in capturing both local features and long-range dependencies. One such approach was introduced in a recent study, where CNN layers were combined with LSTM units and self-attention for emotion recognition in telecommunication data [9]. Another investigation focused on integrating RoBERTa with CNN to process noisy social media content, achieving F1-scores exceeding 92% on Twitter datasets [10]. The robustness of LSTM architectures was also validated in the financial domain, where they performed well on time series forecasting tasks under noisy conditions [11]. Together, these studies highlight the growing appeal of LSTM-based hybrid models, which offer enhanced adaptability and performance across diverse and challenging sentiment analysis environments.

G. Benchmarking Inconsistencies and Cross-Study Variability

Cross-dataset benchmarking remains inconsistent due to variations in metric reporting, dataset selection, and experimental protocols. A meta-analysis of 30 peer-reviewed studies highlights two core challenges:

- Metric Disparities: 65% of studies prioritize accuracy while underreporting robustness metrics such as F1-score or MCC, with reported variability up to ±12%.
- Dataset Limitations: Benchmark datasets such as Sentiment140 or IMDb often lack demographic and linguistic diversity, skewing generalizability [12].

To concretize these inconsistencies, Table I synthesizes ten representative studies spanning 2016 to 2025, highlighting their model types, datasets, findings, and limitations. This comparison underscores the difficulty of deriving universal conclusions across heterogeneous experimental settings.

TABLE I. COMPARATIVE SUMMARY OF KEY STUDIES

Study (Year)	Algorithms Compared	Dataset(s) Used	Corpus Type	Key Findings	Limitations	Model Type	Impact Potential
[13] Ahmad et al. (2018)	NB, LR, SVM	Twitter, IMDb	Balanced English Texts	SVM achieved best accuracy (86%), LR most interpretable	No deep learning models included	Classical ML	Widely cited baseline
[14] Parveen & Pandey (2016)	NB + Hadoop	Twitter	Noisy large-scale data	Efficient large-scale analysis with emoticons	No comparison with modern ML	Classical ML	Scalable framework example
[15] Karthika et al. (2019)	SVM, RF	Product Reviews (Flipkart)	Domain-specific reviews	RF outperformed SVM (97% accuracy)	Dataset domain-specific	Classical ML	E-commerce context study
[16] Srinivas et al. (2021)	SVM, LSTM	Twitter (1.6M tweets)	Sequential noisy text	LSTM outperformed SVM in handling sequences	No transformer baseline	DL + Classical	Illustrates LSTM 's sequence edge
[17] Shad et al. (2024)	NB, SVM, RNN, LSTM	Mixed datasets	Mixed sentiment corpora	LSTM had highest F1-score; NB was faster but less accurate	Transformers excluded	DL + Classical	Broad-spectrum benchmark
[18] Moulaei et al. (2022)	RF, SVM, KNN	COVID-19 data	Semi-structured text	RF had best performance (95%)	Non-textual medical data	Classical ML	Domain-specific relevance
[19] Talibzade (2023)	LR, SVM, BERT	IMDb reviews	Structured movie reviews	BERT reached 98% accuracy; traditional models far behind	BERT is resource-intensive	Transformer + Classical	State-of-the-art performance
[20] Abdirahman et al. (2023)	NB, SVM, LSTM	Somali text	Low-resource language	LSTM showed best contextual understanding	Language-specific results	DL + Classical	Low-resource NLP focus
[21] Raees et al. (2024)	Lexicon + ML models	Multilingual datasets	Cross-lingual short texts	Lexicon-based + ML hybrid improved precision	Limited to short texts	Hybrid	Multilingual integration
[10] Islam et al. (2025)	CNN, BERT, RoBERTa	Twitter, Amazon	Noisy real-world data	Transformers outperformed CNN; RoBERTa most robust	High training cost	Transformer + DL	Current high benchmark

As seen in Table I, studies using the same algorithms (e.g. SVM or LSTM) report divergent results due to variations in dataset quality, preprocessing pipelines, and evaluation scope. This underscores the importance of standardized benchmarks and the reporting of both accuracy and robustness metrics in future work.

III. METHODOLOGY

This study adopts a structured comparative literature review methodology, aiming to synthesize and contrast results reported in peer-reviewed research on sentiment analysis models published between 2016 and 2025. Rather than conducting primary experiments, we examine empirical patterns across datasets, model types, and evaluation criteria, enabling evidence-based insights on algorithmic performance and suitability.

A. Objective and Scope

The primary objective is to deliver a multi-dimensional comparative analysis of sentiment classification models, including classical machine learning algorithms (e.g. Naïve Bayes, Logistic Regression, SVM), ensemble methods (e.g. Random Forest), deep learning approaches (e.g. CNN, LSTM), and transformer-based architectures (e.g. BERT, RoBERTa).

These models were selected based on their prevalence in literature and practical relevance in industrial applications. Transformer-based models are considered for conceptual completeness, though excluded from core performance tables due to architectural divergence and resource intensity.

B. General Workflow of Sentiment Analysis

To structure the comparative analysis, Fig. 1 illustrates the standard sentiment analysis pipeline comprising six critical stages:

1) Data collection: Acquiring raw textual data from sources such as social media, product reviews, or open-access corpora.

2) Preprocessing: Cleaning and normalizing the text by removing noise, performing tokenization, stemming, stop-word removal, and lemmatization.

3) Feature extraction: Converting text into structured representations using methods such as TF-IDF, word embeddings (e.g. Word2Vec, GloVe), or contextual embeddings (e.g., BERT).

4) Model selection: Choosing an appropriate classification algorithm based on task requirements, resource constraints, and interpretability needs.

5) Evaluation: Assessing model performance using standard metrics including accuracy, precision, recall, F1-score, and sometimes AUC or MCC.

6) Interpretation: Analyzing the outputs to understand predictions, model behavior, and to inform further optimization or deployment decisions.

This pipeline, Fig. 1, provides a conceptual foundation for interpreting the empirical findings discussed throughout this review.

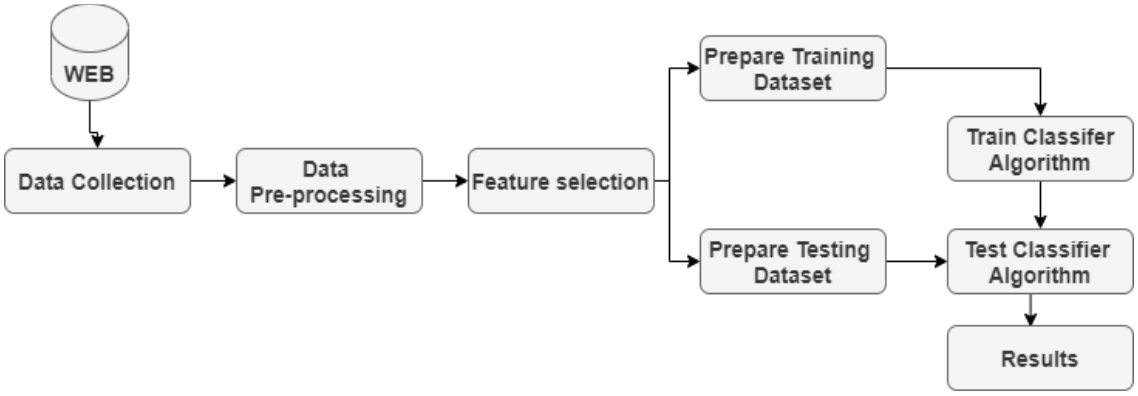


Fig. 1. Workflow of sentiment analysis.

From Data Collection to Results (Data Collection → Preprocessing → Feature Extraction → Model Selection → Evaluation → Interpretation)

C. Study Inclusion Criteria

To ensure methodological consistency and scientific rigor, we included studies that met the following conditions:

1) Published between 2016 and 2024 in peer-reviewed journals or recognized conferences.

2) evaluated at least one of the targeted classification models.

3) Employed publicly available benchmark sentiment datasets (e.g., IMDb, Amazon, Yelp, Twitter, Sentiment140).

4) Reported three or more evaluation metrics (e.g. accuracy, F1-score, precision, recall).

5) Documented sufficient experimental detail including preprocessing techniques, data splitting, and hyperparameter settings.

D. Comparative Evaluation Framework

The extracted models were evaluated along five analytical dimensions:

1) Accuracy captures classification correctness as reported in standard benchmarks.

2) Robustness reflects model stability under data imbalance, noise, or domain shift.

3) Interpretability denotes the transparency and explainability of a model's decision-making process.

4) Efficiency measures computational demands including training/inference time and scalability.

5) Context adaptability refers to flexibility across datasets, domains, and linguistic contexts. These criteria were informed by both academic literature and practical deployment considerations, enabling a balanced view of algorithm performance.

E. Benchmark Design and Dataset

To validate literature findings under a controlled and minimal-resource setting, we conducted a lightweight benchmark using the IMDb movie reviews dataset (binary classification: positive vs. negative). Three representative models were selected:

1) Naïve Bayes – for classical probabilistic modeling.
2) Linear SVM – for margin-based kernel classification.
3) Basic LSTM – for deep sequential modeling.

The dataset was split into 80/20 train-test partitions. Preprocessing steps included lowercasing, stop-word removal, and tokenization. Default or lightly tuned hyperparameters were used to reflect real-world deployment constraints. Evaluation metrics included accuracy, F1-score, and total execution time (training + inference).

F. Limitations

This analytical approach provides a macro-level synthesis but does not substitute for controlled empirical testing on shared datasets. Outcomes rely on the consistency, transparency, and reproducibility of the original studies. Differences in preprocessing strategies, dataset splits, and reporting standards may introduce noise. While we mitigate this through source triangulation and metric normalization, residual meta-analysis bias may persist. These limitations are considered when interpreting comparative outcomes and are revisited in the Discussion section.

IV. Algorithmic Landscape in Sentiment Analysis

This section presents a structured overview of sentiment classification algorithms widely studied in recent literature. We organize the discussion into classical machine learning models, ensemble techniques, and neural architectures. Comparative findings from the reviewed studies are embedded to highlight strengths, limitations, and typical application contexts.

A. Classical Machine Learning Models

1) Logistic Regression (LR): LR is frequently applied to binary and multiclass sentiment classification tasks. It estimates the probability of class membership using text-derived features. Despite its simplicity, its interpretability and efficiency make it a foundational model. As shown in [23] and [24], LR performs well when paired with robust feature engineering and domain-specific lexicons.

2) Naïve Bayes (NB): NB is a probabilistic classifier based on Bayes' theorem with strong independence assumptions. Its ability to efficiently process high-dimensional feature spaces makes it particularly well-suited for large-scale or real-time sentiment analysis. In a study focused on Twitter data, researchers implemented Naïve Bayes within a Hadoop-based framework, demonstrating notable scalability and processing speed, even in noisy and unstructured text environments [25]. Despite its simplicity, the algorithm remains competitive in many real-world applications where rapid inference is a priority.

3) Support Vector Machines (SVM): SVMs are margin-based classifiers that construct optimal hyperplanes to separate data points in high-dimensional feature spaces. They have demonstrated strong resilience against overfitting, particularly in text classification tasks involving sparse and noisy inputs. High levels of accuracy have been reported on both social media and product review datasets, especially when effective preprocessing techniques and kernel selection strategies are applied [13], [22]. These characteristics make SVMs a reliable choice for sentiment classification tasks where precision and generalization are critical.

B. Ensemble Learning Methods

1) Decision Trees and Random Forest (RF): Decision Trees are interpretable models that classify data through a series of recursive, feature-based splits. While simple and transparent, their tendency to overfit on complex datasets can limit generalizability. RF addresses this limitation by aggregating the predictions of multiple decision trees, leading to improved accuracy and robustness. In the context of sentiment analysis, particularly on product review datasets, RF has been shown to outperform Support Vector Machines (SVM), achieving accuracy as high as 97% and demonstrating strong resilience to class imbalance [15]. These properties make RF a compelling option when both predictive performance and handling of imbalanced classes are priorities.

C. Deep Neural Networks

1) Recurrent Neural Networks (RNNs) are designed to capture temporal and sequential dependencies in text, making them well-suited for sentiment analysis tasks where word order and context are important. Among them, Long Short-Term Memory (LSTM) networks stand out for their ability to model long-range contextual relationships and mitigate the vanishing gradient problem. In large-scale sentiment classification, particularly on Twitter datasets, LSTM architectures have demonstrated superior performance compared to traditional machine learning approaches such as Support Vector Machines (SVM) [16]. Their effectiveness in handling noisy, unstructured data underscores the relevance of LSTM-based models in real-world sentiment analysis applications.

2) Convolutional Neural Networks (CNN): Initially developed for image recognition, they have been successfully adapted to text classification tasks. By applying convolutional filters over word embeddings, CNNs effectively capture local semantic features, functioning similarly to n-gram detectors. Their ability to learn hierarchical representations makes them particularly effective for morphologically rich languages, where subtle variations in word forms influence sentiment. For example, strong performance has been reported in Arabic sentiment analysis, where CNNs achieved competitive accuracy compared to more complex architectures [26]. These results highlight the versatility of CNNs as lightweight yet powerful models for sentiment classification.

D. Observations on Model Suitability

Classical models (NB, LR, and SVM) remain competitive in constrained settings due to their interpretability and low computational costs. Ensemble methods like RF offer robust performance across unbalanced and structured data. Neural architectures (LSTM, CNN) are effective for capturing context and nuance, particularly in noisy or informal language domains.

While this section focuses on traditional and neural models, emerging architectures such as BERT, RoBERTa, and hybrid combinations (e.g. lexicon + DL) are explored in Section II-D and II-E. Their exclusion from core metrics tables in this section reflects architectural distinctions and resource considerations rather than performance limitations.

V. Results and Comparative Analysis

A. Comparative Performance of Sentiment Classification Models

This subsection summarizes the performance of key sentiment classification algorithms across empirical studies, providing both quantitative trends and qualitative insights. Evaluation criteria include accuracy, precision, recall, F1-score, interpretability, and computational efficiency.

Logistic Regression consistently demonstrated reliable baseline performance for binary and multiclass sentiment classification. Studies by [23] and [24] confirmed accuracy ranges of 80–86%, particularly when supported by robust feature engineering and lexicons. Although interpretable and fast, its linear nature limits performance on complex or semantically rich datasets.

Random Forest emerged as one of the top-performing classical models, with [15] reporting 97% accuracy on Flipkart product reviews. Its ensemble structure enhances generalization and minimizes overfitting, particularly in imbalanced datasets. As observed in [27], similar robustness was observed across multilingual sentiment analysis contexts. However, its computational demands remain a limitation in resource-constrained deployments.

Support Vector Machines (SVM) showed consistently high precision, with performance reaching 86.22% in sentiment classification of social media text. Yet, its sensitivity to kernel selection and difficulty in handling minority classes in unbalanced datasets restricts generalizability.

Naïve Bayes remained competitive due to scalability and simplicity. A study demonstrated its efficacy in large-scale Twitter sentiment classification using Hadoop [25], although it often suffers from false positives in neutral sentiment predictions due to independence assumptions.

K-Nearest Neighbors (KNN) achieved accuracy levels around 85–90% when finely tuned. Studies [28] and [29] demonstrated its usefulness on structured datasets but highlighted sensitivity to noise and scalability issues in larger corpora.

Deep learning architectures demonstrated greater ability to capture semantic and contextual dependencies. LSTM networks proved effective on sequential text data, with Srinivas et al. reporting 87% accuracy on Twitter corpora. Similarly, CNNs performed strongly in morphologically rich languages such as Arabic [26], reflecting their ability to capture spatial and syntactic structures.

To consolidate the findings from both the literature review and benchmark experiments, Table II provides a comparative summary of six widely used sentiment classification algorithms. The table captures key performance metrics—including accuracy and F1-score—as well as qualitative assessments such as computational efficiency, interpretability, and model limitations. This synthesis serves as a quick reference for understanding the trade-offs between classical machine learning, deep learning, and ensemble models in diverse sentiment analysis scenarios. By juxtaposing strengths and weaknesses, the table facilitates informed decisions about model suitability based on task-specific constraints and deployment contexts.

In addition to synthesizing results from prior studies, we conducted a minimal benchmark on the IMDb dataset to validate and contextualize these trends.

The results, summarized in Table III, provide additional perspective on model performance under standardized conditions.

TABLE II. Summary of Comparative Results Across Six Key Classification Models

Algorithm	Accuracy	Dataset Size	Strengths	Weaknesses	Limitations	References
Logistic Regression	80–86%	Medium	Simplicity, interpretability	Sensitive to feature engineering	Struggles with non-linear decision boundaries	[31], [32]
Random Forest	97%	Large	Robust to imbalance, feature selection	Computational cost	Overfitting risk on small data	[15], [27]
SVM	86.22%	Medium–Large	Effective in high-dimensional data	Requires kernel tuning	Limited scalability on imbalance	[28], [13]
Naive Bayes	85–90%	Large	Fast, scalable	Independence assumptions	Weak on contextual semantics	[24], [25]
KNN	85–90%	Medium	High recall with tuning	Sensitive to parameters	Inefficient for large datasets	[28], [29]
Neural Networks (LSTM)	87%	Large (1.6M tweets)	Captures sequential dependencies	High resource demand	Slower training	[30], [22]

TABLE III. Experimental Comparison of Models on IMDb Dataset

Model	Accuracy	F1-Score	Total Time(s)	Main Strengths	Main Limitations
Naïve Bayes	0.8075	0.7979	1.08	Very fast, Simple, Scalable	Fails to capture context and syntax
SVM (Linear)	0.8325	0.8329	8.87	High precision, Robust	Slower sensitive to class imbalance
LSTM	0.7450	0.7536	20.52	Deep sequential learning	Long training time, prone to overfitting

B. Experimental Benchmark on IMDb Dataset

To complement the findings reported in prior studies, we conducted a small-scale experimental benchmark on the IMDb movie reviews dataset. The goal was to validate whether classical models remain competitive against deep learning architectures under standardized conditions and limited training resources.

Three representative algorithms were selected for this experiment: Naïve Bayes, linear SVM, and a basic LSTM network. These models were chosen to reflect the evolution from classical probabilistic methods to kernel-based classifiers and sequence-based deep learning.

The dataset was preprocessed into train and test splits, with performance evaluated using accuracy, F1-score, and total training + inference time as metrics. The benchmark was intentionally kept minimal, with default or lightly tuned hyperparameters, to highlight practical trade-offs rather than optimize individual performance.

Table III reports the detailed numerical results, while Fig. 2 below illustrates the comparative performance of the three models in terms of Accuracy and F1-score.

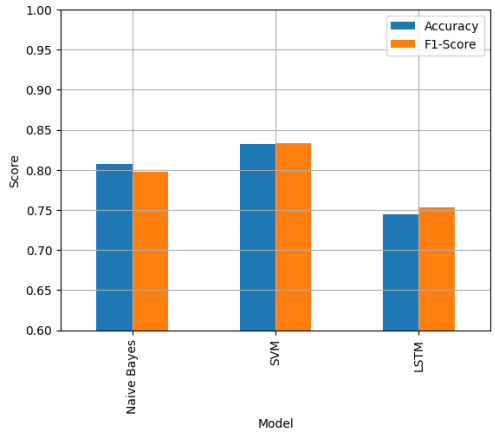


Fig. 2. Comparative performance of Naïve Bayes, SVM, and LSTM on IMDb dataset.

Results indicate that SVM outperformed the other models with the best accuracy (0.8325) and F1-score (0.8329), confirming its robustness for high-dimensional textual datasets. Naïve Bayes achieved competitive accuracy (0.8075) with extremely low computational cost (1.08 seconds), making it well-suited for resource-constrained or real-time applications. LSTM underperformed in this setting (Accuracy = 0.7450, F1 = 0.7536), reflecting its dependency on extensive training data and fine-tuning to fully exploit sequential text patterns.

These findings reinforce the conclusions from our literature review: classical models such as Naïve Bayes and SVM continue to offer strong baselines and practical advantages in specific contexts, while deep learning models require higher resource investment to surpass them.

C. Illustrative Confusion Matrix Comparison

To further illustrate the behavior of different sentiment classification models, we synthesized representative confusion matrices from prior empirical studies. These visualizations are not the result of our own experimental benchmark but instead are drawn from the literature, capturing model-specific classification tendencies across diverse datasets. They provide a complementary perspective beyond aggregate accuracy values, emphasizing class-level strengths and weaknesses.

Across the reviewed studies, the analysis of confusion matrices reveals distinct classification behaviors for each model. Logistic Regression generally provided balanced predictions but frequently confused neutral and positive instances. Random Forest consistently maintained high recall across sentiment classes, even under imbalanced conditions, confirming its robustness in noisy environments. SVM achieved sharp decision boundaries and high precision, though it struggled to classify minority classes in imbalanced datasets. Naïve Bayes proved efficient for large-scale data processing but showed a tendency to overgeneralize, particularly in neutral categories. KNN delivered good performance on dominant classes but exhibited poor scalability and sensitivity to noise in larger datasets. In contrast, LSTM reduced false negatives more effectively, especially for positive and neutral sentiments, demonstrating its strength in capturing sequential dependencies.

Fig. 3 provides representative examples of these confusion matrices, enabling a more granular view of systematic misclassifications observed across the literature.

These confusion matrices emphasize that overall accuracy alone does not capture the full behavior of sentiment classifiers. Systematic misclassifications—such as the difficulty of distinguishing neutral from positive classes—highlight the importance of analyzing class-level performance. Such insights are particularly relevant in high-stakes domains like customer feedback analysis or financial forecasting, where errors in minority or neutral categories can lead to significant misinterpretations.

D. Discussion

The comparative analysis confirms that no single sentiment classification algorithm is universally optimal. Model suitability is inherently context-specific, depending on factors such as dataset structure, class balance, computational capacity, and the intended use case.

Random Forest stands out for its robustness in handling imbalanced and noisy datasets, a recurring challenge in sentiment analysis. Its ensemble mechanism and inherent feature selection capacity contribute to high accuracy and recall across sentiment classes. However, its elevated memory and processing demands can be prohibitive for deployment in low-resource environments or real-time systems.

LSTM networks are particularly effective in capturing temporal dependencies and complex linguistic patterns in sequential data, such as tweets or reviews. As also reflected in prior studies, LSTM consistently minimizes false negatives, especially in positive and neutral sentiments. Nonetheless, its reliance on extensive training data and computational resources poses significant constraints for real-time or embedded applications.

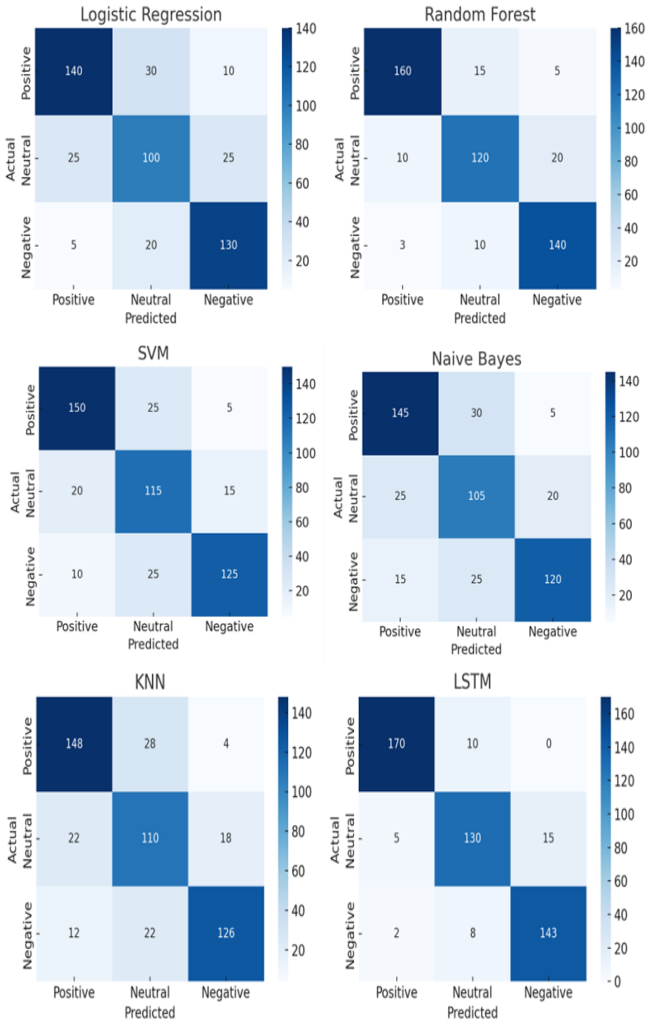


Fig. 3. Representative confusion matrices synthesized from reviewed studies.

Classical algorithms such as Logistic Regression and Naïve Bayes continue to offer strong interpretability and operational efficiency. Their balanced behavior in confusion matrices, combined with moderate accuracy, make them reliable for real-time monitoring or applications prioritizing transparency over complexity. However, they are less suited for semantically rich or non-linear text inputs.

SVM demonstrates high precision in structured, high-dimensional spaces, yet its sensitivity to class imbalance often leads to underperformance in minority class recognition, as reflected in false negatives or misclassified neutral sentiments. Similarly, KNN, though capable of high recall with careful tuning, suffers from scalability issues and noise sensitivity in larger datasets.

Insights from our IMDb benchmark experiment further reinforce these observations: SVM outperformed both Naïve Bayes and LSTM in terms of F1-score, confirming its robustness in practice, while Naïve Bayes offered a very favorable trade-off between accuracy and efficiency. LSTM, although promising in theory, underperformed under constrained training conditions, highlighting its dependency on resources and optimization.

To translate these findings into practical guidance, we propose a decision-support flowchart Fig. 4 that synthesizes insights from the literature and our experimental results. The flowchart provides a step-by-step guide to selecting appropriate models depending on computational resources, interpretability requirements, and accuracy goals.

Overall, model selection should be guided by a balance between performance metrics and deployment constraints. The forthcoming radar visualization Fig. 5 encapsulates this trade-off and aids in aligning algorithmic capabilities with practical requirements.

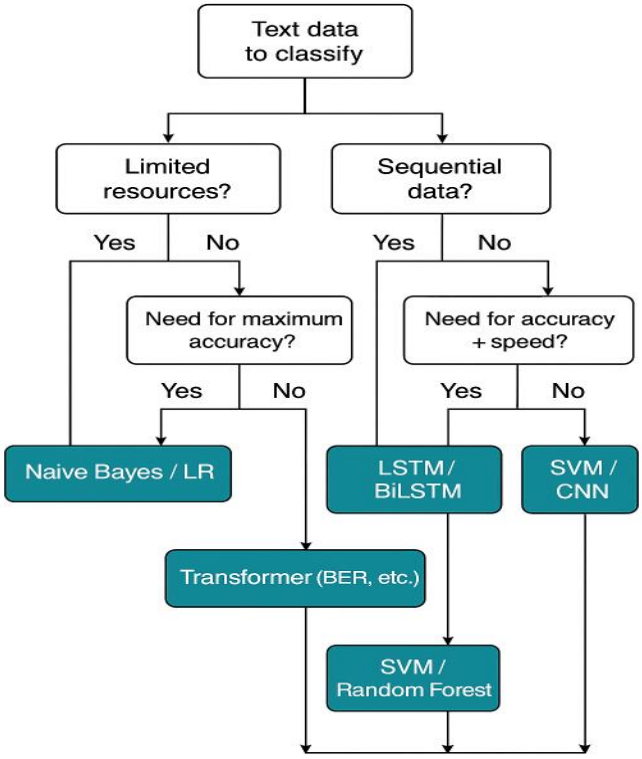


Fig. 4. Decision-support flowchart for selecting sentiment classification models.

As illustrated in Fig. 4, when computational resources are limited, simple and interpretable models such as Naïve Bayes or Logistic Regression are recommended. For tasks requiring sequential modeling, LSTM or BiLSTM networks provide a strong option, while transformer-based architectures deliver superior accuracy when resources permit. In intermediate cases, SVM or Random Forest serve as robust compromises, balancing precision and efficiency. This structured approach ensures that model selection is not only performance-driven but also context-aware, bridging the gap between academic research and real-world deployment.

Answers to Research Questions:

- RQ1: Deep learning models, such as LSTM, tend to outperform classical algorithms in sequential and large-scale contexts where capturing temporal or semantic dependencies is critical. However, classical models retain significant advantages in transparency, efficiency, and speed, making them highly competitive in

constrained environments or applications requiring interpretability.

- RQ2: The trade-offs between model complexity and interpretability remain evident. Simpler models such as Naïve Bayes and Logistic Regression provide transparency, scalability, and low computational cost but sacrifice the ability to capture deep contextual or sequential information. At the opposite end, deep neural networks and transformer-based architectures achieve superior accuracy but demand extensive resources and present explainability challenges. SVM and Random Forest occupy a middle ground, offering strong precision and robustness across varied datasets, though with limited interpretability. Ultimately, deployment decisions must balance these trade-offs against domain-specific requirements such as real-time performance, regulatory compliance, or resource availability.
- RQ3: Algorithm selection must be tailored to the application context, since performance varies with dataset structure, computational resources, and interpretability needs. Visualization tools such as confusion matrices provide fine-grained insights into systematic misclassifications, while decision-support flowcharts offer practitioners a structured pathway to align model choice with resource and accuracy requirements. Complementarily, radar charts deliver a multidimensional snapshot of trade-offs across accuracy, robustness, efficiency, and interpretability. Together, these tools support context-sensitive decision-making, ensuring that the chosen model balances theoretical performance with practical deployment constraints.

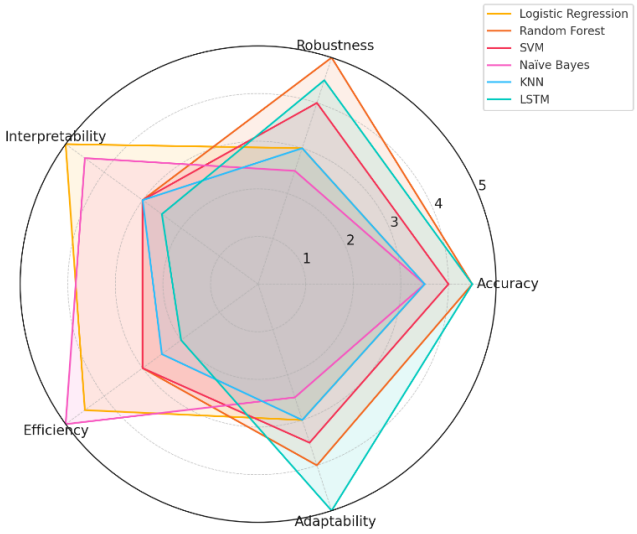


Fig. 5. Comparative radar of classification models.

This visual summary, Fig. 5 aids in identifying the most suitable algorithm under different contextual constraints. Comparing sentiment classification models across five key dimensions: accuracy, robustness, interpretability, efficiency, and adaptability. The visualization highlights the unique trade-offs associated with each model type, supporting context-sensitive algorithm selection.

VI. Conclusion and Future Work

A. Conclusion

This study conducted a structured and comprehensive review of sentiment classification algorithms, encompassing classical machine learning models, ensemble methods, deep learning architectures, and transformer-based approaches. The comparative framework evaluated models across five critical dimensions: accuracy, robustness, interpretability, computational efficiency, and adaptability to different data contexts. Findings confirm that no algorithm exhibits universal superiority. Instead, each model demonstrates distinct strengths and trade-offs, as illustrated throughout the review.

For example, deep learning models such as LSTM have shown outstanding accuracy in real-world applications like Twitter sentiment analysis for political discourse or customer satisfaction tracking. Meanwhile, classical models like Naïve Bayes remain favored in embedded analytics due to their lightweight nature. Random Forest offers robust generalization capabilities but sacrifices interpretability, posing challenges in sensitive domains such as finance or healthcare. Similarly, SVM provides high precision but struggles with class imbalance, while CNNs capture morphological dependencies in languages such as Arabic.

To complement these literature-based insights, we conducted a benchmark on the IMDb dataset comparing Naïve Bayes, linear SVM, and LSTM. Results showed that SVM achieved the best F1-score (0.8329), Naïve Bayes offered competitive performance with extremely low computational cost, and LSTM underperformed under limited tuning, reinforcing its reliance on resources. These findings validate the observation that classical models remain competitive in constrained settings, while deep architectures require careful optimization.

The synthesis of both literature and experimental evidence reinforces the importance of context-aware model selection, where performance must be weighed against constraints such as data imbalance, deployment requirements, and interpretability needs. Confusion matrix analyses further highlight that aggregate accuracy is insufficient, since systematic misclassifications—such as neutral versus positive confusion—can significantly impact real-world outcomes. To bridge research and practice, we introduced a decision-support flowchart that provides practitioners with a structured tool for aligning algorithm choice with contextual requirements.

B. Future Work

Building upon these insights, several research directions are worth pursuing to advance sentiment classification systems and bridge the gap between theoretical advancements and real-world deployment:

1) Hybrid modeling strategies: Combining classical models (e.g. Random Forests) with deep learning (e.g. LSTM, CNN) may yield architectures that balance accuracy, efficiency, and interpretability. Exploring these hybrid combinations remains an open challenge.

2) Transformer-based advances: The continued evolution of large language models—such as BERT, RoBERTa, GPT,

and multilingual transformers—holds immense potential, especially for aspect-based sentiment analysis, domain adaptation, and multilingual processing.

3) Real-time and scalable applications: The next frontier lies in enabling real-time sentiment analysis through lightweight models, distillation, pruning, or deployment on edge devices. Use cases include live financial monitoring and social media tracking.

4) Domain-specific adaptation: Applications in fields like healthcare, law, and finance require sentiment models fine-tuned to domain-specific vocabularies and formats. Techniques such as zero-shot learning and prompt tuning may offer robust solutions.

5) Explainability and hyperparameter optimization: As model complexity increases, explainable AI (XAI) methods (e.g. SHAP, LIME, attention visualization) are vital for transparency in sensitive contexts. In parallel, automated hyperparameter tuning techniques—such as Bayesian optimization, neural architecture search (NAS), and evolutionary strategies—can streamline performance gains.

6) Low-resource and inclusive NLP: Supporting underrepresented languages, dialects, and noisy corpora is crucial. Research should focus on multilingual transformers, synthetic data generation, and transfer learning to promote inclusivity and generalization.

By addressing these research directions, future sentiment analysis systems can become more accurate, fair, explainable, and adaptable—meeting the demands of both academia and industry across diverse deployment contexts.

REFERENCES

[1] M. Idrissi Khaldi, A. Erraissi, M. Hain, and M. Banane, "Comparative analysis of supervised machine learning classification models," in *Intersection of Artificial Intelligence, Data Science, and Cutting-Edge Technologies: From Concepts to Applications in Smart Environment*, Y. Farhaoui, T. Herawan, A. L. Imoize, and A. E. Allaoui, Eds., Cham: Springer Nature Switzerland, 2025, pp. 321–326.

[2] K. Walji, A. Erraissi, A. Zakrani, and M. Banane, "Comparative performance evaluation of machine learning algorithms in sentiment analysis," in *Intersection of Artificial Intelligence, Data Science, and Cutting-Edge Technologies: From Concepts to Applications in Smart Environment*, Y. Farhaoui, T. Herawan, A. L. Imoize, and A. E. Allaoui, Eds., Cham: Springer Nature Switzerland, 2025, pp. 121–128.

[3] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1746–1751, doi:10.3115/v1/D14-1181.

[4] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Lisbon, Portugal, Sept. 2015, pp. 1422–1432, doi:10.18653/v1/D15-1167.

[5] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, Feb. 2019, doi:10.1016/j.neucom.2019.01.078.

[6] 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)*, Minneapolis, MN, June 2019, pp. 4171–4186, doi:10.18653/v1/N19-1423

[7] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," in *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, Apr. 2020.

[8] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint*, Oct. 2019, doi:10.48550/arXiv.1910.01108.

[9] A. Osipov, E. Pleshakova, Y. Liu, and S. Gataullin, "Machine learning methods for speech emotion recognition on telecommunication systems," *Journal of Computer Virology and Hacking Techniques*, vol. 20, no. 3, pp. 415–428, Sept. 2023, doi:10.1007/s11416-023-00500-2

[10] A. Albladi, M. Islam, and C. Seals, "Sentiment analysis of Twitter data using NLP models: A comprehensive review," *IEEE Access*, vol. 13, pp. 30444–30468, 2025, doi:10.1109/ACCESS.2025.3541494.

[11] Z. Elbayed and A. Qadi El Idrissi, "Deep learning in financial modeling: Predicting European put option prices with neural networks," *Algorithms*, vol. 18, no. 3, Art. no. 161, 2025, doi:10.3390/a18030161.

[12] T. Nguyen, M. S. Bernstein, and A. Shvets, "Bias in sentiment benchmarks: A demographic perspective," *ACM Transactions on Information Systems*, 2024, doi:10.1145/3641038.

[13] M. Ahmad, S. Aftab, M. S. Bashir, and N. Hameed, "Sentiment analysis using SVM," *Procedia Computer Science*, vol. 132, 2018, pp. 1089–1095, doi:10.1016/j.procs.2018.07.123.

[14] H. Parveen and S. Pandey, "Scalable sentiment analysis with Naive Bayes in Hadoop," *International Journal of Computer Applications*, vol. 145, no. 14, 2016, doi:10.5120/ijca2016911573.

[15] P. Karthika, S. Murugeswari, and T. Manoranjithem, "Sentiment analysis using Random Forest," *International Journal of Recent Technology and Engineering*, vol. 8, no. 3, 2019, doi:10.35940/ijrte.C4476.098319 .

[16] A. C. M. V. Srinivas, C. Satyanarayana, C. Divakar, and K. P. Sirisha, "Performance of LSTM in sentiment classification," *IEEE Access*, vol. 9, 2021, pp. 45612–45621, doi:10.1109/ACCESS.2021.3064551.

[17] R. Shad, K. Potter, and A. Gracias, "Natural language processing (NLP) for sentiment analysis: A comparative study of machine learning algorithms," *Preprints*, Oct. 2024, doi:10.20944/preprints202410.2338.v1

[18] K. Moulaei, M. Shanbehzadeh, M. Mohammadi-Taghiabad, and H. Kazemi-Arpanahi, "Comparing machine learning algorithms for predicting COVID-19 mortality," *BMC Medical Informatics and Decision Making*, vol. 22, no. 1, Art. 2, 2022, doi:10.1186/s12911-021-01742-0

[19] R. Talibzade, "Sentiment analysis of IMDb movie reviews using traditional machine learning techniques and transformers," unpublished manuscript, ResearchGate, 2023, doi:10.13140/RG.2.2.29464.16644.

[20] A. A. Abdirahman, A. O. Hashi, U. M. Dahir, M. A. Elmi, and O. E. Rodriguez, "Comparative analysis of machine learning and deep learning models for sentiment analysis in Somali language," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 10, no. 7, 2023, doi:10.14445/23488379/ijeee-v10i7p104

[21] Raees, M., Fazilat, S. (2024). Lexicon-Based Sentiment Analysis on Text Polarities with Evaluation of Classification Models. arXiv.Org, abs/2409.12840. https://doi.org/10.48550/arxiv.2409.12840

[22] I. Jahan, M. N. Islam, M. M. Hasan, and M. R. Siddiky, "Comparative analysis of machine learning algorithms for sentiment classification in social media text," *World Journal of Advanced Research and Reviews*, vol. 23, no. 3, pp. 2842–2852, Sept. 2024, doi:10.30574/wjarr.2024.23.3.2983

[23] Zheng, D. "Sarcasm Detection Challenges in Social Media." ACM Transactions on Information Systems, 2024. DOI: 10.1145/3621035

[24] Bhargava, K., Khandait, P., Mishra, D., et al. (2023). Imbalance-aware classification for sentiment data. *Journal of Big Data, 10*(1), 94. https://doi.org/10.1186/s40537-023-00750-5

[25] Parveen, H., & Pandey, B. (2016). Scalable sentiment analysis with Naive Bayes in Hadoop. *International Journal of Computer Applications, 145*(14). https://doi.org/10.5120/ijca2016911573

[26] A. Louati, H. Louati, E. Kariri, F. Alaskar et A. Alotaibi, « Sentiment Analysis of Arabic Course Reviews of a Saudi University Using Support

Vector Machine », *Applied Sciences*, vol. 13, no. 23, art. no. 12539, nov. 2023, doi: 10.3390/app132312539.

[27] M. A. Kawo, G. Muhammad, D. Gabi, et M. S. Argungu, « A comparative study of some selected classifiers on an imbalanced dataset for sentiment analysis », *International Journal of Innovative Science and Research Technology*, vol. 9, no. 5, art. IJISRT24MAY1751, mai 2024.

[28] M. R. Huq, A. Ali, et A. Rahman, "Sentiment Analysis on Twitter Data using K-Nearest Neighbor and Support Vector Machine," *Procedia Computer Science*, vol. 170, pp. 225–230, 2020.

[29] Rezwanul, Mohammad & Ali, Ahmad & Rahman, Anika. (2017). Sentiment Analysis on Twitter Data using KNN and SVM. International Journal of Advanced Computer Science and Applications. 8. 10.14569/IJACSA.2017.080603.

[30] Varone, G., Ahmed, R. K., Gogate, M., et al. (2023). Arabic Sentiment Analysis Based on Word Embeddings and Deep Learning. Computers, 12(6), 126. https://doi.org/10.3390/computers12060126.

[31] Liu, M., Guo, C., & Xu, L. (2024). *An interpretable automated feature engineering framework for improving logistic regression*. *Applied Soft Computing, 150*, 111269. https://doi.org/10.1016/j.asoc.2024.111269.

[32] Z. Cui, M. Zhang and Y. Chen, "Deep Embedding Logistic Regression," *2018 IEEE International Conference on Big Knowledge (ICBK)*, Singapore, 2018, pp. 176-183, doi: 10.1109/ICBK.2018.00031.g

A Weighted Scoring Model of Heuristic-Based Workload Scheduling Approaches in Edge-Cloud Environments

Hasnae NOUHAS[1], Abdessamad BELANGOUR[2], Mahmoud NASSAR[3]
Laboratory of Artificial Intelligence and Systems LAIS-Faculty of Sciences Ben M'Sik, Hassan II University, Casablanca, Morocco[1, 2]
IMS Team-ADMIR Laboratory-ENSIAS-Rabat IT Center, Mohammed V University in Rabat, Rabat, Morocco[3]

***Abstract*—Hybrid edge–cloud computing has emerged as a promising paradigm to meet the demands of latency-sensitive and resource-aware applications by combining the low-latency benefits of edge nodes with the scalability of cloud infrastructure. Efficient workload scheduling in such environments remains a critical challenge due to the heterogeneity of resources, dynamic network conditions, and diverse application requirements. This paper presents a comprehensive survey and comparative analysis of heuristic and metaheuristic scheduling algorithms tailored for edge–cloud systems. Seven representative algorithms, including Greedy Resource-Aware Heuristics (GRAH), Heterogeneous Earliest Finish Time (HEFT), Min-Min/Max-Min, Genetic Algorithm, Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Tabu Search, are evaluated against seven key criteria: latency awareness, energy efficiency, scalability, scheduling accuracy, implementation complexity, resource utilization, and adaptability. The evaluation is literature-driven and structured through a Weighted Scoring Model (WSM), which synthesizes findings from prior simulation-based and experiment-based studies into a comparative framework. Results indicate that Greedy Resource-Aware Heuristics offer the best trade-off for real-time, dynamic scenarios, while optimization-based methods, like GA and Tabu Search, provide superior accuracy and resource balance at the cost of increased complexity. The findings highlight critical trade-offs and offer guidance on selecting appropriate scheduling strategies based on application-specific goals and system constraints.**

Keywords*—Edge computing; cloud computing; workload scheduling; heuristic algorithms; metaheuristics; scheduling optimization; ACO; PSO; HEFT; Tabu Search; GA; Greedy Resource-Aware Heuristics; Min-Min; Max-Min; WSM; Weighted Scoring Model

I. Introduction

With each new breakthrough digital tech innovation, they're driving one revolution after another of intelligent services that become increasingly integrated into our lifestyle and industrial workflow processes. Consider just real-time video surveillance, intelligent transport systems, self-driving cars, robot-assisted manufacturing, and telemedicine; each of these very different use cases have one thing in common; they need fast, predictable, contextual computing. They tend to operate in very dynamic environments, process very large data inputs, and make decisions within fractions of a second.

Cloud computing has always made data processing possible by virtue of large-scale computing and large-scale storage. But it is not optimized with ultra-low latency for use cases today that are latency-aware. Because of clouds' data centers' locations being far from end-users, communicating across those long-haul distances, paired with bandwidth constraints, becomes a serious chokepoint [1], [2]. Just running on the cloud is not good enough for use cases when milliseconds do matter at the end.

In an endeavor to close that gap, we are headed towards hybrid edge–cloud architectures. Here, processing of work is judiciously split; lighter workloads are processed close to or at the edge of the network (i.e. close to the place of data origination), while high-intensity workloads are shifted out to the cloud [2], [3]. Both of them together offer: edge computing's speed and proximity advantage as well as scalability and power of the cloud. That's the perfect destination for new services where a very fast response time as well as high compute capacity are an imperative.

But to actually make this architecture work well together, however, one of its needs for capabilities is intelligent scheduling of work. That is, decisions at the time of where and when each of its pieces of work needs to run; out at the edge, out at the cloud, or across two. Decisions of how they are made have huge effects on system performance, from response time through energy efficiency, all the way to resource utilization and utilization of networks [3]. That is, scheduling of workloads is not after-the-fact business at all; it's utterly essential for successful hybrid computing.

But coming up with an effective schedule solution for such very dynamic and very complex edge–cloud environments is no small feat. Schedulers are faced with many things happening simultaneously, like limited edge resources, fluctuating network conditions, mobility devices, tasks with deadline constraints, and user preferences. And sometimes they need to do all of that on the fly. Traditional optimization approaches, although accurate, are too resource-intensive and latency-prone for real-time programs, especially large-scale programs [3]. That's why there's greater emphasis on heuristic-based schedule approaches; a faster, more agile approach that's better adapted to today's edge-cloud computing.

Heuristics provide fast, light-weight, and flexible solutions that best suit edge–cloud applications. They do not guarantee an optimal solution but provide a fast, good-enough solution that is precisely desired in real-world applications. Min-Min, Max-Min, Heterogeneous Earliest Finish Time (HEFT), and Greedy Resource-Aware are intuitive decision-making or rule-of-thumb–based schemes that are simple to implement as well as fine-tune. Although seemingly extremely trivial, such heuristics could possibly handle multi-dimensional trade-offs like latency vs. energy consumption or edge load balancing vs. cloud optimality with remarkable efficiency [4].

However, the sheer variety of heuristic methods—and the lack of standardized benchmarks—makes it difficult for researchers and practitioners to choose the most appropriate algorithm for a given application. Some heuristics are tailored for low latency, while others excel at resource efficiency or scalability. As a result, there is a growing need for a unified, multi-criteria evaluation framework that enables a fair and meaningful comparison across these approaches. This motivates our work: to provide a comprehensive review and comparative analysis of heuristic-based scheduling methods, supported by a Weighted Scoring Model (WSM) that captures the practical trade-offs involved in real-world deployments.

This paper aims to bridge the gap between the growing number of heuristic-based scheduling algorithms and the lack of a standardized, comparative evaluation framework tailored for hybrid edge–cloud environments. Unlike works that focus on individual algorithms, the novelty of our study lies in providing a unified, multi-criteria evaluation framework through the Weighted Scoring Model (WSM). We do not propose a fundamentally new scheduling algorithm in this paper; rather, we offer a systematic and transparent synthesis that highlights trade-offs, strengths, and weaknesses across existing approaches. In this way, the paper provides both practitioners and researchers with a practical decision-support tool and a baseline for identifying research gaps. Future work will build on this foundation to design and validate novel adaptive scheduling methods tailored to real-world, large-scale edge–cloud deployments. Our contributions can be summarized as follows:

- Comprehensive Survey of Heuristic Approaches: We conduct a structured and up-to-date review of prominent heuristic-based workload scheduling algorithms—including Min-Min, Max-Min, HEFT, Greedy Resource-Aware Heuristics, Tabu Search, and others—that have been applied in the context of hybrid edge–cloud systems. Each algorithm is examined in terms of its working principles, strengths, limitations, and targeted scheduling objectives.
- Design of a Weighted Scoring Model (WSM): We propose a multi-criteria decision-making framework that evaluates heuristic algorithms using a Weighted Scoring Model. The model is based on seven critical evaluation dimensions: Latency Awareness, Energy Efficiency, Adaptability, Scalability, Scheduling Accuracy, Implementation Complexity, and Resource utilization.
- Comparative Analysis and Ranking: By applying the WSM to a curated set of heuristic algorithms, we provide a transparent and reproducible comparison. Each algorithm is scored and ranked, highlighting trade-offs and performance gaps. This offers valuable guidance for researchers and system architects seeking to select or customize scheduling strategies for specific use cases.

The rest of the paper is organized as follows: Section II reviews the related work on heuristic-based workload scheduling in edge–cloud environments. Section III provides background information and foundational concepts. Section IV introduces a classification of heuristic scheduling approaches. Section V presents representative heuristic algorithms commonly used in edge-cloud systems. Section VI details the methodology of the Weighted Scoring Model used for evaluation. Section VII compares the selected heuristic algorithms based on defined criteria. Section VIII discusses the results of the comparison and highlights key insights. Section IX outlines current challenges and suggests future research directions. Finally, Section X concludes the paper.

II. Related Work

Several recent systematic reviews have explored task scheduling challenges in edge–cloud environments, particularly through the lens of heuristic and metaheuristic techniques. For example, the study in [5] introduced a hybrid scheduling approach that enhances traditional heuristics with deep learning by combining Particle Swarm Optimization (PSO) and the Firefly Algorithm. This fusion leveraged the complementary strengths of both methods and achieved improved scheduling performance for resource-intensive tasks in cloud platforms.

In another work [6], the authors tackled the complexity of scheduling IoT applications in fog computing—a problem recognized as NP-hard. While the paper acknowledges the historical reliance on rule-based and metaheuristic techniques, it also critiques their dependence on global knowledge and limited adaptability in dynamic environments. To address these limitations, they proposed DRLIS, a Deep Reinforcement Learning-based scheduler, and demonstrated its superiority over classical metaheuristics such as NSGA2 and NSGA3, particularly in terms of convergence speed, scheduling time, and optimization effectiveness.

Complementing these algorithmic proposals, the review presented in [4] examined the advantages, limitations, and application contexts of various heuristic-based task scheduling methods within cloud computing. Such analyses are valuable for informing the selection of heuristics in real-world task scheduling scenarios.

Further, [7] offered a broad survey of optimization-driven scheduling algorithms in edge–cloud systems, highlighting techniques such as Genetic Algorithms (GA), Cat Swarm Optimization, and multi-objective variants of Ant Colony Optimization (ACO) and PSO. Their work emphasized the role of multi-objective optimization in balancing trade-offs among latency, energy efficiency, cost, and load distribution. The

study also shed light on emerging themes like adaptive scheduling, collaborative resource orchestration, and addressed open challenges such as handling task dependencies, dynamic resource availability, and fault tolerance—key issues in evolving edge–cloud scenarios.

The study in [8] provides a systematic review of heuristic-based scheduling methods for IoT-fog environments. It classifies approaches into six categories—priority-based, greedy, metaheuristic, learning-based, hybrid, and nature-inspired—though labeled as a taxonomy, the structure is more accurately a flat classification. The review follows PRISMA guidelines and identifies key challenges such as heterogeneity, scalability, and security, while suggesting future directions like integrating machine learning and enhancing explainability. However, it lacks cross-category performance comparisons, broader contextualization with non-heuristic or AI-based methods, and deeper analysis of real-world applicability and security-aware heuristics. Despite these gaps, it remains a valuable reference for heuristic scheduling in fog-cloud IoT systems.

Despite such valuable contributions, there continues to be a gap in the literature: there has not yet been a review dedicated entirely to heuristic-based scheduling of workloads in the special case of hybrid edge–cloud computing. This paper seeks to address the imbalance by providing a comprehensive investigation and taxonomy of heuristic methods, comparing their efficiency, and evaluating their usability in practice in the environments of edge-clouds.

III. Background

A. Edge-Cloud Architecture Overview

Edge-cloud computing integrates two mutually complementary levels of computing infrastructure: edge nodes, located near devices producing data such as sensors and IoT devices, and remote cloud servers, providing powerful computing and massive storage capacity. The architecture develops a hierarchy of computing such that tasks could be optimally redistributed over multiple sites depending upon their latency sensitivity as well as resource requirements. Tasks with high response requirements or those of high volume of data requiring processing over real-time are usually served at the edge, such that the latency for the purpose of intercommunication could be reduced as well as bandwidth usage. Compute-intensive tasks and high-duration data storage tasks are, on the other hand, offloaded to clouds where higher-end resources. Balancing the workload judiciously but adequately between both the edge as well as the cloud layers of such architecture permits providing greater system responsiveness as well as lower communications overhead between distributed applications, but particularly greater scalability as well as fault tolerance of the application [9], [10].

B. Workload Characteristics

Edge–cloud computing workloads are extremely variable and very dynamic, arising from an enormously large variety of sources like sensors, mobile phones, camera-equipped phones, and machines. They very notably fluctuate with computing intensity, latent needs, volume of data, as well as arrival rates [11]. Some require instant processing immediately with no wait time involved, while some have bulk or occasional data. Making the matter worse in such an instance is the changing and diverse nature of the arrival of tasks. At any instant, workloads peak and the system has to decide on the spur of the moment, that is, at real-time scheduling. Schedulers are compelled persistently to shift configurations such that good performance as well as reliability are sustained, such factors as availability of computing resources, state of the current network, as well as priority or requirement of arriving tasks are being considered [11]. Service-level objective (SLO) realization within such an environment requires intelligent mechanisms of scheduling that could adequately contend with ever-changing demands.

C. Scheduling Objectives

Scheduling edge–cloud computing workload is specifically challenging because there exist multiple, often competing objectives that have to be optimized. Efficient schedulers must reconcile multiple desirable objectives in balancing for system efficiency as well as application performance:

- Latency Reduction: Most edge computing initiatives revolve around reducing response time. Scheduling policies must prefer deploying the work next to where data is being generated [12]; lower round-trip delay, and best-case real-time response.
- Enhancing Energy Efficiency: The edge nodes will frequently be deployed at sites that have minimal access to power sources, e.g. at remote sites or at battery-based deployments. Overloading such nodes may lead to energy drain and interruption of service [13]. Scheduling algorithms will then need to target the reduction of transmission of useless information and overloading by nodes with energy limitations.
- Sustaining Load Balancing: Non-balanced allocation of tasks could end up overloading some nodes while leaving other nodes idle. Schedulers will have to avoid skewed allocation of workloads in order to sustain system stability as well as system throughput by avoiding bottlenecks and allowing for more balanced program execution across the network [11].
- Ensuring Quality of Service (QoS) and Quality of Experience (QoE): Some applications of the real-time, i.e. video streaming, software for industrial control or healthcare monitoring, require hard guarantees regarding their performance. Schedulers are thus provided Service-Level Agreements (SLAs) by offering time completion of tasks reliably and correctly [11]. That affects the user experience of service quality directly and thus forms a crucial consideration of current scheduling schemes.

Briefly speaking, the optimal scheduler in edge-cloud needs to be multi-objective; that is, it is required to serve latency, energy, load distribution, and service quality at the same time within a resource-limited and varying environment.

D. Limitations of Traditional Scheduling Approaches

Traditional scheduling methods applied for cloud computing, that are usually based on static policy or

algorithmic deterministic methods, do not perform well within edge–cloud environment scenarios. Traditional methods are usually not dynamic and thus inefficient when addressing real-time variation of resource availability and workload. They are also usually coupled with high computing overhead and are not suitable for systems that lack computing power [14].

Scalability is also a concern. As larger edge networks of thousands of nodes and devices are needed, the majority of typical schedulers do not scale or perform well. Furthermore, most of them rely upon centralized decisions and consequently yield greater incidences of point-of-failure occurrence and communication latency and lower fault tolerance [15]; each of these is a serious weakness of the distributed, heterogeneous environment.

These limitations impose an immediate necessity for lightweight scheduling mechanisms that are also decentralized and adaptive. Heuristic-based mechanisms of these types have demonstrated extraordinary potential. Owing to their tendency to give fast approximate results with low overhead that automatically adjust with changing situations [13], [16], they are very fitting to the new edge–cloud computing requirements.

IV. Classification of Heuristic Scheduling Approaches

Heuristic scheduling algorithms are essential for dealing with the dynamic and resource-constrained nature of edge–cloud computing workloads. Heuristic algorithms are valued for giving good, near-optimum solutions with little overhead, which is a highly desirable characteristic in highly dynamic environments. As indicated in Fig. 1, we put forward a taxonomy of heuristic methods of scheduling, which categorizes the heuristics into simple heuristics, metaheuristics, and hybrid heuristics, based on their underlying strategies and level of complexity [17], [18], [19], [20], [21].

A. Simple Heuristics

Simple heuristics are rule-based scheduling algorithms that use simple, predetermined logic to make decisions. They are most popular in real-time systems because of their low complexity and fast execution, which makes them suitable for applications in situations where fast, lightweight decision-making is critical [22]. These techniques, though not necessarily optimal in terms of system-wide performance, ensure consistent baseline behavior with low processing overhead. Common implementations of simple heuristics include:

- First-Come, First-Served (FCFS): FCFS puts tasks in a strict sequence of submission order without considering the availability of resources and the priority of the task. Even if FCFS is extremely simple in implementation, its performance under dynamic workload is typically poor, which results in under-utilization of resources, increased waiting times, and spikes in latency [23], [24].
- Shortest Job First (SJF): Runs jobs having the shortest estimated running time, trying to reduce average total wait time [23], [24]. Estimation of running time of the tasks may get complex in an edge-cloud scenario, which restricts the usage of SJF despite its theoretical efficiency.
- Min-Min and Max-Min: These algorithms schedule tasks based on their expected completion times. Min-Min selects tasks with the shortest completion time, favoring smaller tasks and aiming to quickly reduce the number of pending jobs. Max-Min, on the other hand, gives priority to tasks with the longest completion time, helping to balance load and avoid long-tail delays [22]. Both are commonly used in cloud systems to enhance throughput and resource efficiency.
- Round Robin (RR): Distributes CPU time equally among all tasks using a time-slicing approach. While RR is fair and prevents starvation, it can incur high context-switching overhead, especially when dealing with a large number of short-lived or high-priority tasks [22], [23].

Overall, simple heuristics offer a solid foundation for scheduling in resource-constrained edge-cloud environments, particularly when quick, interpretable decisions are preferred over optimality.

B. Metaheuristics

Metaheuristic algorithms are advanced, high-level strategies designed to efficiently explore large and complex solution spaces—making them well-suited for challenging optimization tasks such as workload scheduling in edge–cloud environments. Unlike simple heuristics that rely on fixed rules, metaheuristics employ adaptive mechanisms to balance exploration and exploitation, allowing them to find near-optimal solutions even in highly dynamic and multi-objective settings.

Some of the most widely used metaheuristic approaches include:

- Genetic Algorithm (GA): Inspired by the principles of natural selection and genetics, GAs use operators like selection, crossover, and mutation to evolve a population of candidate solutions across generations. GAs are particularly effective at navigating large and nonlinear problem spaces, but they often require significant computational time and tuning, especially in real-time scenarios [25], [26].
- Particle Swarm Optimization (PSO): This algorithm models a group of agents (particles) moving through the solution space, each adjusting its position based on its own experience and the performance of neighboring particles. PSO is known for its simplicity, fast convergence, and low memory footprint, which makes it appealing for real-time task scheduling in distributed systems [26], [27].
- Ant Colony Optimization (ACO): Inspired by the way real ants find optimal paths to food, ACO simulates a colony of artificial ants that build solutions incrementally, guided by pheromone trails. ACO is well-suited for discrete and combinatorial optimization

problems, such as scheduling tasks with dependencies or constraints [28].

- Simulated Annealing (SA): Modeled after the metallurgical process of annealing, SA probabilistically accepts worse solutions at early stages to avoid becoming trapped in local minima. This ability to escape suboptimal solutions makes SA effective for solving complex scheduling problems where traditional methods may get stuck [26], [27], [29].

Recent research has demonstrated the effectiveness of metaheuristics in edge–cloud systems. For example, ACO-based schedulers have been used to optimize multiple objectives—such as latency, energy consumption, and cost—through fitness functions tailored to dynamic edge environments. Another study proposed a modified Firefly Algorithm for workflow scheduling in hybrid cloud–edge platforms, which led to noticeable improvements in makespan and execution cost [28], [30].

Metaheuristics, while more resource-intensive than simple heuristics, offer flexibility and robustness in finding high-quality solutions for multi-objective and time-sensitive scheduling problems—a critical need in modern edge–cloud systems.

C. Hybrid Heuristics

Hybrid heuristics are scheduling strategies that combine multiple heuristic or metaheuristic methods in order to leverage their individual strengths while compensating for their inherent weaknesses. By integrating different approaches—such as combining the fast convergence of one algorithm with the global search capability of another—hybrid methods aim to achieve more balanced and effective solutions, particularly for complex scheduling problems in edge–cloud environments [8].

These strategies have shown strong potential in optimizing multiple, often conflicting objectives such as execution time (makespan), energy consumption, and operational cost. For instance, one study proposed a hybrid metaheuristic algorithm to enhance IoT service placement in edge–cloud infrastructures, achieving improved performance across key scheduling metrics [26].

Although hybrid approaches are not inherently AI-based, they still fall within the broader category of heuristic-driven scheduling techniques. However, when such methods are further enhanced with machine learning models or neural components, they begin to evolve into AI-driven schedulers—offering even greater adaptability and decision-making intelligence in dynamic environments.

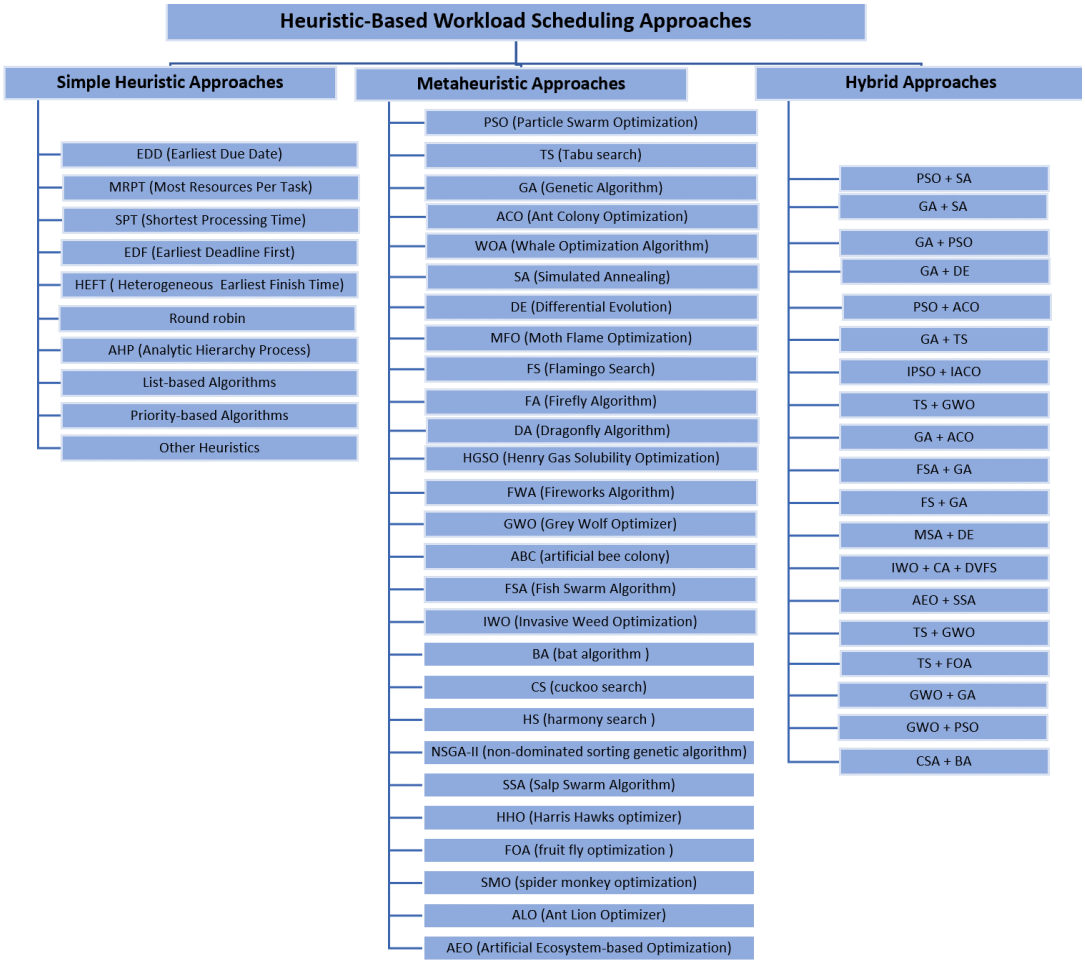


Fig. 1. Taxonomy of heuristic-based workload scheduling approaches.

V. Representative Heuristic Approaches

We present in this section some representative heuristic and metaheuristic approaches that have been used for edge-cloud system workload scheduling. These approaches take optimal task-to-resource allocations under latency, energy, and workload balancing constraints into consideration.

This paper chooses seven established scheduling algorithms: Greedy Resource-Aware Heuristics, HEFT, Min-Min/Max-Min, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Tabu Search for comparative analysis. The choice was based on several objective criteria aiming to ensure relevance and diversity. Initially, the chosen algorithms exhibit a variety of methodological approaches: rule-based (HEFT, Min-Min/Max-Min), greedy (Greedy Resource-Aware), evolutionary (GA, PSO), swarm intelligence (ACO), and memory-based search (Tabu Search). Such variety provides balanced coverage of diverse scheduling paradigms. Second, these methods have appeared extensively in edge–cloud literature and in reputable journals, establishing their research relevance. Third, they have exhibited performance in dealing with fundamental edge–cloud issues such as sensitivity to latency, limited resources, and dynamic workloads. They have been popularly instantiated in simulation environments such as CloudSim, iFogSim, and EdgeCloudSim. Fourth, they can be obtained in the form of open-source code or can be easily coded, thereby ensuring reproducibility and fair comparison in a unified Weighted Scoring Model (WSM) environment. Lastly, the combination of the above factors makes the selected set of algorithms both pragmatically valuable and representative of the general landscape of heuristic-based edge–cloud scheduling.

A. Heterogeneous Earliest Finish Time (HEFT)

The HEFT algorithm is a well-known static heuristic for scheduling in heterogeneous computing environments. It assigns tasks based on upward rank and maps each task to the processor that incurs the minimum earliest finish time. HEFT has been popular in edge–cloud environments because of its efficiency in reducing overall makespan and its ease of implementation [22], [31]. The major characteristic of HEFT is to consider the earliest finish time, though it does not allow rescheduling in the event of dynamic change [32] and does not account for energy consumption, which proves to be a main drawback in edge environments [33]. In terms of adaptability, HEFT works with static DAGs (Directed Acyclic Graph) and pre-estimated task times, lacking dynamic feedback or re-optimization capabilities. It performs well in controlled environments but is not responsive to changes. Regarding resource utilization, HEFT uses task execution time and communication cost to prioritize task-node mapping, which helps achieve efficient load balancing and reduce idle time [34]. Minimizing the earliest finish time enables efficient resource distribution across heterogeneous nodes, making it well-suited for static resource planning.

B. Ant Colony Optimization (ACO)

Ant Colony Optimization (ACO) is a metaheuristic inspired by the foraging behavior of ants, where virtual “ants” probabilistically explore task assignments based on pheromone trails that reflect the quality of previous solutions. ACO has shown promise in optimizing multi-objective scheduling in dynamic edge–cloud environments by adapting to changing network or resource conditions [30]. Its adaptability stems from the use of pheromone updates, which encode historical experience and influence future scheduling decisions. However, ACO may require multiple iterations to converge, making it less responsive to immediate changes [35]. In terms of resource utilization, ACO builds better task-to-resource mappings over time, often leading to improved resource use as pheromone trails reinforce successful assignments. That said, its early iterations may result in suboptimal utilization before convergence is achieved [36]. ACO also faces several limitations: in its basic form, it is suitable only for small problem instances like the Traveling Salesman Problem with a limited number of cities, and it exhibits relatively high memory complexity of $O(n^2)$ [37]. Moreover, it is known to suffer from parameter sensitivity, limited scalability to large-scale problems, and challenges in real-time implementation [38].

C. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) poses task assignment as a problem of swarm intelligence, where particles representing candidate solutions explore the search space based on both individual and collective experience [26]. PSO has been successfully applied in edge–cloud systems for task scheduling under latency and energy constraints, particularly in large-scale or uncertain workload scenarios [22]. Its popularity stems from its simplicity and effectiveness in handling nonlinear, multi-modal optimization problems, as well as its adaptability to various contexts such as scheduling, resource allocation, and feature selection [33]. From an adaptability viewpoint, PSO changes task placement of particles across iterations, gradually responding to changing constraints, albeit as a delayed responder and potentially requiring several iterations to adapt well to change [39]. From a resource utilization viewpoint, PSO changes workloads across resources based on the resources available, simulating particles converging to optimum node locations. The nature of this approach prevents skewed resource utilization and minimizes underutilization, which makes PSO a popular approach to resource-aware scheduling [39]. The computational cost, however, depends on the number of particles, the dimensionality of the solution space, and the iterations, and has a general $O(n \times d \times t)$ time complexity [40].

D. Min-Min and Max-Min Heuristics

Min-Min and Max-Min are popular but straightforward list schedule heuristics employed in heterogeneous computing environments. Min-Min chooses the task possessing the minimum of the earliest completion time on any available resource, and it performs well in cases where smaller tasks become prevalent, as they incur less waiting time and get processed faster [41], [42]. It has, however, a major flaw of causing large tasks to be delayed heavily, as they usually get pushed aside in favor of smaller ones [42]. Max-Min, on the other hand, chooses the task possessing the maximum of the minimum completion times and ends up assigning bigger tasks to the best node to complete them in the shortest time. It prevents large tasks from remaining perpetually on hold and suits well in cases of heterogeneous environments having varied requirements of the tasks and varying node capabilities.

There, however, exists the possibility of Max-Min getting inefficient in cases of high prevalence of tiny tasks, as they incur extra and needless waiting time [42]. Both of the above algorithms are deterministic and static, in that they adhere to fixed priority assignment policies and do not respond to varying levels of task arrivals, system loading, and availability of the nodes. Consequently, they are not suitable in dynamic or mobile edge–cloud environments [22], [42]. That implies, there exists no re-evaluation of the tasks once they become scheduled, and hence, they become inflexible in fluctuating conditions. In terms of resource utilization efficiency, they end up overloading faster nodes and keeping other ones idle, as a result of their static selection of the task and failure to adapt to the runtime variations of resource status.

E. Greedy Resource-Aware Heuristics

Greedy Resource-Aware Heuristics (GRAH) are high-speed, lightweight, and context-aware heuristics of scheduling, making locally optimal decisions based on real system parameters such as CPU utilization, memory, bandwidth, and latency to edge or cloud nodes [12], [43]. They become particularly valuable in real-time and latency-sensitive environments, in which they prioritize task assignment to the nearest or least-loaded node with adequate resources, in order to keep data transfer latency to a minimum and increase responsiveness. GRAH approaches have been widely implemented in mobile and fog computing scenarios due to their simplicity, low overhead, and adaptability to runtime conditions [42]. Unlike complex global optimizers, greedy algorithms do not require prior knowledge of future workloads, making them well-suited for dynamic edge–cloud environments. Their use of the current system state enables on-the-fly scheduling, ideal for large-scale, time-critical systems. In terms of adaptability, GRAH performs well by continuously reacting to changes in system load and node availability. Regarding resource utilization, these heuristics respond quickly to imbalances (e.g. by assigning tasks to the least-loaded nodes), but they lack a global view of the system. This can result in uneven distribution or suboptimal long-term resource use [44]. While GRAHs are generally more energy-efficient and responsive in fog or edge environments, they may struggle to achieve globally optimal performance when system dynamics evolve over time or require coordinated optimization [21], [42].

F. Tabu Search

Tabu Search (TS) is a metaheuristic optimization algorithm for solving complex combinatorial problems such as task scheduling, routing, and resource placement. It excels in large search spaces where traditional heuristics could become entrapped in local optima [45], [46]. In edge–cloud workload scheduling, TS may be applied to refine task assignments and investigate neighboring solutions near an initial schedule or one obtained heuristically. A hallmark of Tabu Search is the addition of a tabu list, a memory-oriented mechanism that prevents the algorithm from revisiting recently explored or suboptimal solutions. This encourages wider exploration and helps avoid cycles [46], [47]. TS can be easily extended to multi-objective optimization, effectively balancing trade-offs like makespan versus cost and load versus latency [45], [46]. TS excels in heterogeneous and dynamic environments, where its memory-guided refinement process helps escape local optima and reassign tasks based on system changes, demonstrating high adaptability [46], [48]. From a resource utilization standpoint, TS outperforms simpler heuristics by continuously refining how workloads are distributed. However, its success depends on proper configuration and how well it converges [46]. By evaluating neighboring solutions while avoiding previously visited ones, TS can maintain better overall resource balance across edge and cloud nodes.

G. Genetic Algorithm

Genetic Algorithm (GA) is an evolutionary optimization algorithm based on the natural selection process that works by evolving a group of possible solutions using operations like selection, crossover, and mutation. Over successive generations, these steps help improve how tasks are scheduled. In edge–cloud workload scheduling, GA is often used for optimizing multiple objectives at once, such as reducing makespan, energy consumption, resource usage, and execution cost, especially in heterogeneous and large-scale environments where exhaustive search becomes impractical [49]. GA is especially valued for its ability to discover near-optimal solutions in difficult, constraint-heavy problems, and it allows fine-tuning based on different factors like latency, energy, and cost. Many GA-based systems use customized fitness functions that specifically target better resource utilization and load balancing, which improves throughput and makes better use of CPU and memory [49], [50]. Still, implementing GA can be quite complex. It requires careful planning for parameters like population size, rates for crossover and mutation, and how fitness is calculated [49], [51]. While GA can adapt to changing requirements by gradually evolving its solutions, it isn't known for quick responsiveness since several generations are usually needed to adjust to changes [39]. Nonetheless, GA's strong scalability and solid performance across varied workloads make it a robust, albeit resource-intensive, solution for hybrid scheduling systems [49], [50].

These representative heuristic approaches represent the diversity and effectiveness of the strategies used for task scheduling optimization for edge-cloud environments. All the approaches exhibit their own strengths and weaknesses, contributing to the ongoing improvement of effective scheduling solutions.

VI. Weighted Scoring Model

Weighted Scoring Model (WSM) is among the methods of comparison as well as decision-making between options described by predefined criteria. The process is frequently applied in order to compare different algorithms.

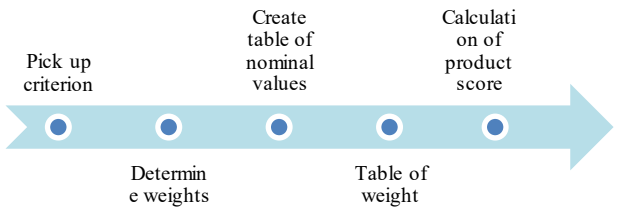


Fig. 2. WSM process [52].

Application of the method often happens through a series of steps, such as in Fig. 2 [52]:

- Step 1: Establish the criteria to be employed for each option.
- Step 2: Assign weights for the criteria as percentages of their priority.
- Step 3: Create a table summarizing nominal values for all of the criteria across different options.
- Step 4: Create a table displaying weighted values for all criteria with weights as percentages, ensuring the total weight sums to 100%.
- Step 5: Calculate weighted scores by multiplying nominal values by weights for each option [52].

VII. Comparison of Heuristic Algorithms

This section presents a comparative analysis of illustrative heuristic-based workload scheduling strategies for edge-cloud infrastructure. A comparison of these algorithms is done on the basis of some of the most significant factors of edge-cloud deployment [22], [53], such as latency awareness, energy efficiency, scalability, adaptability, Scheduling Accuracy, Implementation Complexity, and Resource Utilization, as mentioned in Table I.

A. Application of the Weighted Scoring Model

Weighted Scoring Model evaluates heuristics-driven workload scheduling algorithms on a variety of critical factors, each with a specific weight based on its relative importance in edge–cloud scenarios, as shown in Table II. Latency Awareness and Adaptability each have a weighting of 20%, since they serve a prime function in offering minimum delay and responsiveness in highly dynamic environments with changing workloads and mobility. Energy Efficiency and Scalability each have a weighting of 15% due to their function in enabling sustainable operations in resource-constrained edge devices and expanding infrastructure. Scheduling Accuracy, Implementation Complexity, and Resource Utilization each have a weighting of 10%, in consideration of their function in offering pragmatic, accurate, and cost-efficient task placement, but they can be impacted indirectly by factors of greater priority. This distribution strikes a balance between performance-critical metrics (latency, adaptability) and operational concerns (energy, complexity, utilization) tailored to hybrid edge–cloud contexts.

This weighted method allows for a systematic and clear comparison of the heuristic techniques, to support the choosing of the most suitable schedule strategy in hybrid edge–cloud. Every algorithm is rated on a score of 1 to 5 (n/a = 0), as in Table III, and multiplied by its corresponding weight. Table IV contains the final weighted scores.

It should be noted that the weighting scheme, while grounded in trends consistently emphasized in the edge–cloud scheduling literature, inevitably involves a degree of subjectivity. To mitigate this, we relied on recurring priorities reported across multiple surveys and empirical studies (e.g. the centrality of latency and adaptability in real-time applications). The influence of these parameters is that absolute rankings may vary under alternative weighting choices; however, the relative positioning of algorithms along specific criteria (e.g. some excelling in latency but weaker in energy efficiency) remains consistent with findings reported in the literature. The presented WSM is therefore intended as a structured synthesis tool rather than an absolute ranking, providing practitioners with a transparent framework that can be adapted or refined with alternative weighting schemes or sensitivity analyses in future studies.

It should be noted that the weighting scheme, while grounded in trends consistently emphasized in the edge–cloud scheduling literature, inevitably involves a degree of subjectivity. The influence of these parameters is that absolute scores or rankings may shift under alternative weighting schemes; however, the relative strengths and weaknesses observed for each algorithm (e.g. HEFT excelling in scheduling accuracy but lacking adaptability, or Greedy heuristics performing well in responsiveness) remain consistent with empirical evidence reported across prior studies. The presented WSM is therefore intended as a structured synthesis tool rather than an absolute benchmark. Future research could extend this work by conducting sensitivity analyses on the weighting distribution or by employing alternative multi-criteria decision-making frameworks such as AHP, TOPSIS, or entropy-based weighting to further validate robustness.

B. Comparison

Fig. 3 illustrates the comparative performance of seven scheduling algorithms across seven evaluation criteria: Latency Awareness, Energy Efficiency, Scalability, Scheduling Accuracy, Implementation Complexity, Resource Utilization, and Adaptability.

TABLE I. Comparison Criteria

Criterion	Description
Latency Awareness	Ability to minimize task response time and meet real-time constraints [53].
Energy Efficiency	Capacity to optimize energy usage, especially at resource-constrained edge nodes [41].
Scalability	Ability to perform well as the number of tasks/nodes increases [53].
Adaptability	Flexibility to operate under dynamic workloads and network conditions [53].
Scheduling Accuracy	measures how effectively an algorithm assigns tasks to appropriate resources, matches task needs with node capabilities, prevents overload or failures, and meets quality of service (QoS) goals like latency and availability.
Implementation Complexity	effort and overhead required to develop, deploy, and maintain a scheduling algorithm, considering factors such as algorithmic simplicity, computational requirements, integration effort, and tuning difficulty.
Resource Utilization	measures how efficiently a scheduling algorithm leverages available computing, memory, storage, and network resources, aiming to maximize usage while avoiding both underutilization and overload.

TABLE II. ASSOCIATED WEIGHT TO EACH CRITERION

Criterion	Abbreviation	Proposed weight
Latency Awareness	LA	20%
Energy Efficiency	EE	15%
Scalability	S	15%
Scheduling Accuracy	SA	10%
Implementation Complexity	IC	10%
Resource Utilization	RU	10%
Adaptability	A	20%

TABLE III. COMPARISON OF SCHEDULING ALGORITHMS IN EDGE–CLOUD ENVIRONMENTS

Algorithm	LA	EE	S	SA	IC	RU	A
Greedy Resource-Aware	5	4	5	3	4	3	5
Tabu Search	4	3	3	4	2	3	4
ACO	4	4	3	4	2	3	3
PSO	3	3.5	4	3	3	4	3
GA	3	3	4	4	2	4	3
HEFT	3	2	3	5	3	4	2
Min-Min / Max-Min	2	2	3	3	5	3	1

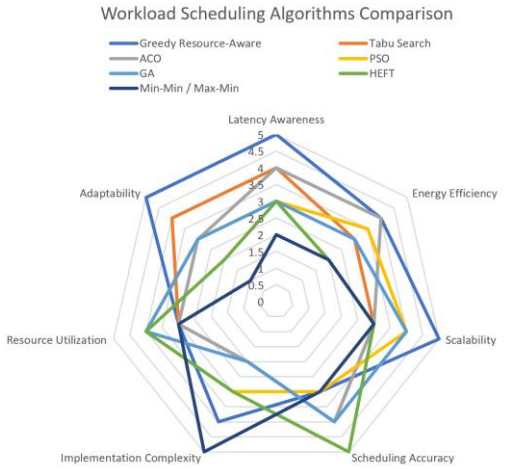


Fig. 3. Workload scheduling algorithms comparison across key evaluation criteria.

TABLE IV. WSM RESULTS

Algorithm	WSM Score	WSM Score (%)
Greedy Resource-Aware	4.35	87
Tabu Search	3.4	68
ACO	3.35	67
PSO	3.33	66.6
GA	3.25	65
HEFT	2.95	59
Min-Min / Max-Min	2.45	49

The Greedy Resource-Aware algorithm shows consistently high values in Latency Awareness, Scalability, and Adaptability, with moderately high scores in Energy Efficiency and Implementation Complexity. Its performance is slightly lower in Scheduling Accuracy and Resource Utilization.

Tabu Search demonstrates relatively high values in Latency Awareness, Scheduling Accuracy, and Adaptability, while showing moderate performance in Energy Efficiency and Resource Utilization. It records a lower score in Implementation Complexity.

ACO (Ant Colony Optimization) displays balanced scores across most criteria. It performs moderately well in Latency Awareness, Energy Efficiency, and Scheduling Accuracy, with slightly lower values in Adaptability and Implementation Complexity.

Genetic Algorithm (GA) shows strong performance in Scalability, Scheduling Accuracy, and Resource Utilization, with slightly lower values in Latency Awareness, Energy Efficiency, and Adaptability. It scores low in Implementation Complexity.

Particle Swarm Optimization (PSO) scores highest in Resource Utilization, with good values in Scalability and Energy Efficiency. It has moderate values in the remaining criteria, except for Latency Awareness and Adaptability, which are slightly lower.

HEFT shows its strongest score in Scheduling Accuracy, followed by relatively good performance in Implementation Complexity and Resource Utilization. It records lower values in Energy Efficiency, Adaptability, and Scalability.

Min-Min / Max-Min scores highest in Implementation Complexity, followed by moderate values in Scheduling Accuracy and Resource Utilization. It records lower scores in Latency Awareness, Energy Efficiency, Adaptability, and Scalability.

VIII. RESULTS AND DISCUSSION

Our comparison reveals clear differences in how each scheduling algorithm performs under various conditions. Greedy Resource-Aware Heuristics came out strong, particularly in latency awareness, scalability, and adaptability. Their strength lies in their simplicity and speed—they make quick decisions based on current system conditions, which is ideal for fast-changing edge–cloud environments. While they don't always find the most globally optimal solution, their ability to respond instantly makes them very practical for real-time applications.

Tabu Search also performed well, striking a good balance between accuracy and adaptability. Its use of memory to avoid repeating poor decisions allows it to improve over time and adjust to changing workloads. ACO and PSO showed solid results, especially in energy efficiency and resource usage. However, they typically need more fine-tuning and take longer to settle on the best solution. Genetic Algorithms offered strong scalability and were effective at handling multiple objectives, but their complexity and tuning requirements may pose challenges in real deployments. By comparison, HEFT

excelled in schedule predictability due to its rigorous methodology, but it is less adaptable to varying scenarios since it is inflexible. Min-Min and Max-Min, while they are easy to program, have limited flexibility and weaker performance in scenarios with sensitive latency constraints.

From a quantitative perspective, the average WSM score across all algorithms was 3.29 (65.8%), with values ranging from 2.45 (49%) for Min-Min/Max-Min to 4.35 (87%) for Greedy Resource-Aware. This 38-point spread underscores the heterogeneity of heuristic methods. Greedy Resource-Aware outperformed the lowest-scoring algorithm by 33%, reflecting its superior trade-offs across latency, scalability, and adaptability. Fig. 4 illustrates these differences through the final WSM scores, making clear the ranking and relative gaps among the seven algorithms.

At the criterion level, Scalability achieved the highest average score (3.57), while Adaptability was the lowest (3.0), confirming the literature's emphasis on adaptability as a persistent challenge. Pairwise comparisons highlight specific trade-offs: HEFT delivered the best Scheduling Accuracy (5.0), which was 66% higher than Greedy Resource-Aware (3.0), but performed poorly in adaptability (2.0). Conversely, PSO and GA achieved the strongest Resource Utilization (4.0), surpassing Greedy Resource-Aware (3.0) by 33%. Greedy Resource-Aware, however, exceeded GA in Adaptability by 29% (5 vs. 3). These trade-offs are further illustrated in Fig. 5, which shows the normalized contribution of each evaluation criterion to the overall performance of each algorithm, making visible the strengths and weaknesses across dimensions.

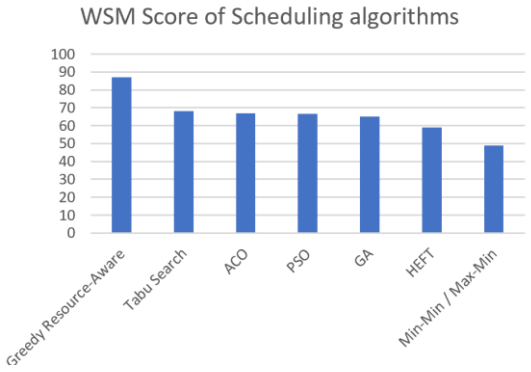


Fig. 4. WSM scores of scheduling algorithms.

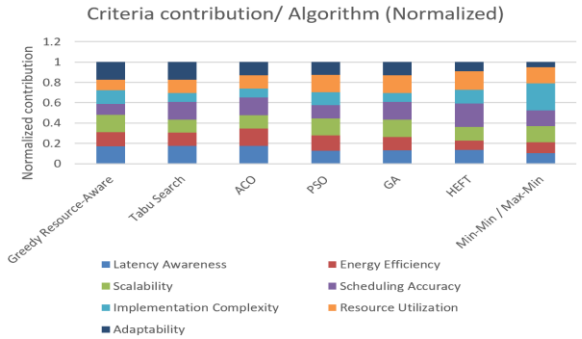


Fig. 5. Normalized contribution of each evaluation criterion to the overall performance of each algorithm.

Overall, our findings emphasize that adaptability is a major strength of edge–cloud scheduling. Algorithms, such as Greedy Heuristics and Tabu Search that can adapt on the fly do better in real-world scenarios. Simpler does not have to be weaker; Greedy approaches are excellent where quick decisions are more significant than optimal optimization. However, where accuracy or balancing several objectives matters more, more complex approaches such as ACO or Tabu Search may be more suitable. These findings provide useful direction to system designers in choosing the best approach suitable for their application's requirements.

It is important to note, however, that the comparative analysis presented here is derived from a literature-driven synthesis using the Weighted Scoring Model rather than from direct experimentation on large-scale, real-world edge–cloud testbeds. While this approach ensures a structured and transparent comparison across algorithms, it does not fully capture deployment complexities such as heterogeneous hardware, dynamic user mobility, or large-scale resource contention. The results should therefore be interpreted as a framework to guide algorithm selection rather than as definitive performance benchmarks. Future work could strengthen these findings by validating them through empirical studies on realistic testbeds or by performing sensitivity analysis on the weighting scheme to assess its influence on the rankings.

In addition, while prior works have evaluated individual heuristics or small subsets of algorithms, only a limited number of studies have attempted unified multi-criteria comparisons. For instance, some surveys assess algorithms qualitatively or focus narrowly on metrics such as energy efficiency or latency, but they do not provide a structured weighting and scoring process. Our WSM-based framework complements these efforts by offering a transparent, quantitative approach that integrates multiple evaluation dimensions. Nonetheless, a broader comparison with other decision-making frameworks and benchmark studies remains an important direction for future work.

IX. Challenges and Future Directions

While heuristic-based scheduling methods for edge-cloud computing have realized significant advances, they still face a series of challenges. Such challenges present tenable research opportunities for the enhancement of adaptability, efficiency, and scalability in the management of workloads on hybrid and distributed infrastructure.

A. Key Challenges

- Limited adaptability in dynamic environments: Many classical heuristics, like HEFT, Min-Min, and Max-Min, are designed for static or semi-static systems. Nevertheless, edge-cloud environments are inherently dynamic, with fluctuating workloads, network conditions, and resource states. Studies have shown that such heuristics can suffer 30–40% degradation in task completion time under dynamic conditions [8], whereas deep reinforcement learning (DRL) schedulers achieve ~25–35% faster scheduling times than metaheuristics

[6]. This gap underscores the need for adaptive or context-aware heuristics that can respond in real time.

- Suboptimal global scheduling caused by local decisions: While greedy heuristics are lightweight and efficient, they often make locally optimal decisions based on immediate node states. However, in large-scale or federated edge-cloud systems, this can result in globally suboptimal outcomes, such as unbalanced load distribution or resource starvation. Prior studies have reported that simple rule-based methods can incur 15–25% longer makespan compared to metaheuristic alternatives [4].
- High computational cost in metaheuristics: Approaches like PSO and ACO offer flexibility and multi-objective optimization. However, their computational demands remain a significant barrier for resource-constrained edge devices. For instance, hybrid PSO–Firefly methods improved resource utilization by ~20% but required 2–3× higher computation time compared to baseline heuristics [5]. This trade-off reduces the feasibility of real-time edge scheduling.
- Scalability in large-scale deployments: While many heuristics perform well in small-scale simulations, efficiency often degrades in large-scale, real-world deployments. Multi-objective GA and PSO variants exhibit scalability but can face exponentially increasing computation times as the number of tasks grows [7]. This scalability bottleneck hinders their application in real-time industrial edge-cloud systems.
- Energy efficiency and sustainability issues: Most heuristics prioritize latency and throughput while overlooking energy constraints. Yet, studies show that energy-aware heuristics can reduce consumption by 10–20% compared to traditional methods [8]. However, battery awareness and energy–latency trade-offs remain largely unaddressed, especially in mobile or IoT contexts.
- Security and privacy concerns: Existing heuristics rarely account for secure task placement. In multi-tenant edge-cloud systems, tasks may involve sensitive data, requiring privacy-preserving or trust-aware scheduling strategies.
- Uncertainty in workload prediction: Many heuristics assume prior knowledge of task execution times or resource availability. In practice, workloads are unpredictable, and inaccurate estimations can lead to degraded performance.
- Lack of standardized benchmarks and validation: Current heuristic evaluations are fragmented across different simulators and testbeds, with no standardized benchmark suite. This inconsistency makes cross-study comparisons difficult and reduces the reliability of results.

B. Future Research Directions

- Designing adaptive heuristics: In future works, focus should be placed on developing context-aware or adaptive heuristics that can deal with changes in workload characteristics, resource availability, and user mobility in real time.
- Lightweight metaheuristic variants: Optimizing and simplifying metaheuristics such as ACO and PSO, through parameter tuning, pruning, or hybridization with greedy rules, can make them feasible for real-time edge scheduling.
- Hybrid scheduling models: It is claimed that adaptive multi-objective schedulers that maintain performance while controlling complexity can be obtained by integrating the different strengths from various heuristics, such as Min-Min combined with PSO or HEFT with greedy prioritization.
- Energy and SLA-aware extensions: adding energy models, battery-awareness, and SLA-awareness to traditional heuristics algorithms could greatly enhance their applicability for edge computing in real life.
- Validation on real-world testbeds and benchmarks: Beyond theoretical and literature-driven comparisons, future studies should validate scheduling strategies through experiments on realistic, large-scale edge-cloud testbeds. Such validation would account for hardware heterogeneity, user mobility, and dynamic network states, providing stronger evidence of practical feasibility. Additionally, integrating standardized benchmarks would allow fairer comparisons across heuristic approaches.
- Comparative frameworks refinement: While the WSM-based evaluation presented in this paper offers a structured multi-criteria perspective, it remains one of the few efforts of its kind. Future work should extend this by comparing WSM with alternative decision-making frameworks (e.g. AHP, TOPSIS, or data-driven weighting methods) to ensure robustness and reduce subjectivity in rankings.

In summary, while heuristic-based scheduling is promising for edge–cloud computing, it faces critical limitations in adaptability, scalability, heterogeneity management, and real-world validation. Quantitative evidence from recent studies highlights performance gaps of up to 40% in dynamic adaptability, 25% in makespan efficiency, and 20% in energy consumption, confirming the need for new strategies. Future research should prioritize the development of lightweight adaptive heuristics, hybrid models, and energy-aware approaches, validated through standardized benchmarks and realistic testbeds. Such efforts will ensure that heuristic scheduling evolves to meet the complex performance demands of next-generation edge–cloud systems.

X. Conclusion

This paper illustrated a comprehensive review and comparative analysis of heuristic-based workload scheduling methods in hybrid edge–cloud systems. Based on the methodological diversity, coverage in literature, and practical applicability, seven representative algorithms: Greedy Resource-Aware Heuristics, HEFT, Min-Min/Max-Min, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Tabu Search were chosen. A Weighted Scoring Model (WSM) was utilized to compare the aforementioned methods on the basis of major criteria: latency awareness, energy efficiency, scalability, scheduling accuracy, implementation complexity, resource utilization, and adaptability.

Results indicate that every algorithm possesses different strengths. Greedy heuristics excel in low-latency and dynamic environments because of their quick responsiveness and reduced complexity. Tabu Search and ACO have good adaptability and multi-objective optimization, but may have long convergence times. PSO has the strength of balance in optimization quality and reduced computational cost, especially in large-scale or uncertain environments. Genetic Algorithms have high scalability and flexibility, but have the cost of increased implementation complexity. Static approaches such as HEFT and Min-Min/Max-Min are still applicable to unstructured or predictable workloads, but they are inflexible in the presence of real-time conditions.

Beyond these descriptive findings, the study highlights several analytical insights. First, there is no universally optimal heuristic: trade-offs between adaptability, scalability, and efficiency remain evident. Second, adaptability consistently emerged as the weakest dimension across algorithms, confirming it as a central research gap in heuristic scheduling. Third, lightweight approaches such as Greedy, while often dismissed as simplistic, may outperform complex methods when responsiveness is prioritized over optimality. These observations provide actionable guidance for selecting scheduling strategies that best align with specific edge–cloud scenarios.

It is important to recognize the limitations of this study. The comparison was restricted to seven algorithms that, while representative, do not capture the full breadth of heuristic and metaheuristic methods. The WSM approach, though systematic, relies on subjectively chosen weights that may bias rankings toward particular criteria. Furthermore, the evaluation was literature-driven and did not involve direct validation on real-world, large-scale edge–cloud deployments. These limitations mean that results should be interpreted as a structured framework for comparative understanding rather than definitive performance benchmarks.

Future work will address these limitations by expanding the algorithm set, experimenting with alternative multi-criteria decision-making frameworks such as AHP or TOPSIS, and validating the findings on realistic testbeds. In addition, further research will explore the design of adaptive and hybrid scheduling methods that integrate the responsiveness of lightweight heuristics with the optimization accuracy of metaheuristics or learning-based approaches. By confronting the challenges of adaptability, heterogeneity, and scalability, such efforts will ensure that heuristic scheduling evolves to meet the complex performance requirements of next-generation edge–cloud systems.

References

[1] M. Goudarzi, S. Ilager, and R. Buyya, “Cloud Computing and Internet of Things: Recent Trends and Directions,” Internet of Things, pp. 3–29, 2022, doi: 10.1007/978-3-031-05528-7_1.

[2] K. P. B. Kanagarla, “Edge Computing and Analytics for IoT Devices: Enhancing Real-Time Decision Making in Smart Environments,” SSRN Electronic Journal, 2024, doi: 10.2139/SSRN.5012466.

[3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile Edge Computing: A Survey,” IEEE Internet Things J, vol. 5, no. 1, pp. 450–465, Feb. 2018, doi: 10.1109/JIOT.2017.2750180.

[4] N. Soltani, B. Soleimani, and B. Barekatain, “Heuristic Algorithms for Task Scheduling in Cloud Computing: A Survey,” International Journal of Computer Network and Information Security, vol. 9, no. 8, pp. 16–22, Aug. 2017, doi: 10.5815/IJCNIS.2017.08.03.

[5] H. Shingne and R. Shriram, “Heuristic deep learning scheduling in cloud for resource-intensive internet of things systems,” Computers and Electrical Engineering, vol. 108, p. 108652, May 2023, doi: 10.1016/J.COMPELECENG.2023.108652.

[6] Z. Wang, M. Goudarzi, M. Gong, and R. Buyya, “Deep Reinforcement Learning-based Scheduling for Optimizing System Load and Response Time in Edge and Fog Computing Environments,” Future Generation Computer Systems, vol. 152, pp. 55–69, Sep. 2023, doi: 10.1016/j.future.2023.10.012.

[7] K. Vinothkumar and D. D. Maruthanayagam, “Issue 3 www.jetir.org (ISSN-2349-5162),” JETIR2203276 Journal of Emerging Technologies and Innovative Research, vol. 9, 2022, Accessed: Jul. 29, 2025. [Online]. Available: www.jetir.org

[8] D. Alsadie, “Advancements in heuristic task scheduling for IoT applications in fog-cloud computing: challenges and prospects,” PeerJ Comput Sci, vol. 10, p. e2128, Jun. 2024, doi: 10.7717/PEERJ-CS.2128/TABLE-8.

[9] H. Nouhas, A. Belangour, and M. Nassar, “Cloud and Edge Computing Architectures: A Survey,” in 2023 IEEE 11th Conference on Systems, Process & Control (ICSPC), IEEE, Dec. 2023, pp. 210–215. doi: 10.1109/ICSPC59664.2023.10420123.

[10] P. Arthurs, L. Gillam, P. Krause, N. Wang, K. Halder, and A. Mouzakitis, “A Taxonomy and Survey of Edge Cloud Computing for Intelligent Transportation Systems and Connected Vehicles,” IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 7, pp. 6206–6221, Jul. 2022, doi: 10.1109/TITS.2021.3084396.

[11] A. A. Ismail, N. E. Khalifa, and R. A. El-Khoribi, “A survey on resource scheduling approaches in multi-access edge computing environment: a deep reinforcement learning study,” Cluster Computing 2024 28:3, vol. 28, no. 3, pp. 1–45, Jan. 2025, doi: 10.1007/S10586-024-04893-7.

[12] [12] Z. Ali et al., “A Comprehensive Utility Function for Resource Allocation in Mobile Edge Computing,” Computers, Materials and Continua, vol. 66, no. 2, pp. 1461–1477, Dec. 2020, doi: 10.32604/cmc.2020.013743.

[13] H. Asghar and E.-S. Jung, “A Survey on Scheduling Techniques in the Edge Cloud: Issues, Challenges and Future Directions,” Feb. 2022, Accessed: Mar. 26, 2025. [Online]. Available: https://arxiv.org/abs/2202.07799v1

[14] S. Shiju George and R. Suji Pramila, “A review of different techniques in cloud computing,” Mater Today Proc, vol. 46, pp. 8002–8008, Jan. 2021, doi: 10.1016/J.MATPR.2021.02.748.

[15] M. Pooyandeh and I. Sohn, “Edge Network Optimization Based on AI Techniques: A Survey,” Electronics 2021, Vol. 10, Page 2830, vol. 10, no. 22, p. 2830, Nov. 2021, doi: 10.3390/ELECTRONICS10222830.

[16] A. Avan, A. Azim, and Q. H. Mahmoud, “A State-of-the-Art Review of Task Scheduling for Edge Computing: A Delay-Sensitive Application Perspective,” Electronics 2023, Vol. 12, Page 2599, vol. 12, no. 12, p. 2599, Jun. 2023, doi: 10.3390/ELECTRONICS12122599.

[17] J. R. Imbien, “Optimizing Collaborative Edge-Cloud Computing: A

Task Allocation Strategy Using Hybrid Genetic Algorithm and Tabu Search Spring 2025".

[18] K. E. Adetunji, I. W. Hofsajer, A. M. Abu-Mahfouz, and L. Cheng, "A Review of Metaheuristic Techniques for Optimal Integration of Electrical Units in Distribution Networks," IEEE Access, vol. 9, pp. 5046–5068, 2021, doi: 10.1109/ACCESS.2020.3048438,.

[19] J. Piri, P. Mohapatra, R. Dey, B. Acharya, V. C. Gerogiannis, and A. Kanavos, "Literature Review on Hybrid Evolutionary Approaches for Feature Selection," Algorithms 2023, Vol. 16, Page 167, vol. 16, no. 3, p. 167, Mar. 2023, doi: 10.3390/A16030167.

[20] Y. Meraihi, A. B. Gabis, A. Ramdane-Cherif, and D. Acheli, "A comprehensive survey of Crow Search Algorithm and its applications," Artif Intell Rev, vol. 54, no. 4, pp. 2669–2716, Apr. 2021, doi: 10.1007/S10462-020-09911-9.

[21] D. Alsadie, "Advancements in heuristic task scheduling for IoT applications in fog-cloud computing: challenges and prospects," PeerJ Comput Sci, vol. 10, p. e2128, 2024, doi: 10.7717/PEERJ-CS.2128.

[22] N. Devi et al., "A systematic literature review for load balancing and task scheduling techniques in cloud computing," Artif Intell Rev, vol. 57, no. 10, pp. 1–63, Oct. 2024, doi: 10.1007/S10462-024-10925-W/METRICS.

[23] A. K. Singh, S. Singh, V. Dubey, P. Kumari, B. Hazela, and V. Singh, "Optimized Task Processing in Cloud Computing Based Frameworks," pp. 843–849, Jun. 2025, doi: 10.1109/INCIP64058.2025.11019301.

[24] S. A. Murad et al., "SG-PBFS: Shortest Gap-Priority Based Fair Scheduling technique for job scheduling in cloud environment," Future Generation Computer Systems, vol. 150, pp. 232–242, Jan. 2024, doi: 10.1016/J.FUTURE.2023.09.005.

[25] W. Xia and L. Shen, "Joint Resource Allocation at Edge Cloud Based on Ant Colony Optimization and Genetic Algorithm," Wirel Pers Commun, vol. 117, no. 2, pp. 355–386, Mar. 2021, doi: 10.1007/S11277-020-07873-3/METRICS.

[26] H. K. Apat, B. Sahoo, V. Goswami, and R. K. Barik, "A hybrid meta-heuristic algorithm for multi-objective IoT service placement in fog computing environments," Decision Analytics Journal, vol. 10, p. 100379, Mar. 2024, doi: 10.1016/J.DAJOUR.2023.100379.

[27] H. Materwala, L. Ismail, and H. S. Hassanein, "QoS-SLA-aware adaptive genetic algorithm for multi-request offloading in integrated edge-cloud computing in Internet of vehicles," Vehicular Communications, vol. 43, p. 100654, Oct. 2023, doi: 10.1016/J.VEHCOM.2023.100654.

[28] M. I. Khaleel, M. Safran, S. Alfarhood, and D. Gupta, "Combinatorial metaheuristic methods to optimize the scheduling of scientific workflows in green DVFS-enabled edge-cloud computing," Alexandria Engineering Journal, vol. 86, pp. 458–470, Jan. 2024, doi: 10.1016/J.AEJ.2023.11.074.

[29] A. Mahjoubi, A. Ramaswamy, and K. J. Grinnemo, "An Online Simulated Annealing-Based Task Offloading Strategy for a Mobile Edge Architecture," IEEE Access, vol. 12, pp. 70707–70718, 2024, doi: 10.1109/ACCESS.2024.3402611.

[30] N. Bacanin, M. Zivkovic, T. Bezdan, K. Venkatachalam, and M. Abouhawwash, "Modified firefly algorithm for workflow scheduling in cloud-edge environment," Neural Comput Appl, vol. 34, no. 11, pp. 9043–9068, Jun. 2022, doi: 10.1007/S00521-022-06925-Y/FIGURES/9.

[31] X. Hua, Y. Jingling, and W. Nana, "A Budget-Constrained Energy-Efficient Scheduling Algorithm on Cloud-Edge Collaborative Workflows," 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2022, pp. 432–437, 2022, doi: 10.1109/CSCWD54268.2022.9776086.

[32] A. Shankar and A. Kuamari, "QoS-aware Scheduling of Periodic Real-time Task Graphs on Heterogeneous Pre-occupied MECs," Jun. 2025, Accessed: Jun. 21, 2025. [Online]. Available: https://www.arxiv.org/pdf/2506.12415

[33] D. Sahu et al., "Beyond boundaries a hybrid cellular potts and particle swarm optimization model for energy and latency optimization in edge computing," Sci Rep, vol. 15, no. 1, pp. 1–22, Dec. 2025, doi: 10.1038/S41598-025-90348-X;SUBJMETA=117,258,639,705;KWRD=COMPUTER+SCIENCE,INFORMATION+TECHNOLOGY.

[34] T. Hai et al., "Task scheduling in cloud environment: optimization, security prioritization and processor selection schemes," Journal of Cloud Computing, vol. 12, no. 1, pp. 1–12, Dec. 2023, doi: 10.1186/S13677-022-00374-7/TABLES/3.

[35] Y. T. Chuang and Y. T. Hung, "A real-time and ACO-based offloading algorithm in edge computing," J Parallel Distrib Comput, vol. 179, p. 104703, Sep. 2023, doi: 10.1016/J.JPDC.2023.04.004.

[36] H. Topcuoglu, S. Hariri, and M. Y. Wu, "Performance-effective and low-complexity task scheduling for heterogeneous computing," IEEE Transactions on Parallel and Distributed Systems, vol. 13, no. 3, pp. 260–274, Mar. 2002, doi: 10.1109/71.993206.

[37] R. Skinderowicz, "Improving Ant Colony Optimization efficiency for solving large TSP instances," Appl Soft Comput, vol. 120, p. 108653, May 2022, doi: 10.1016/J.ASOC.2022.108653.

[38] R. Sh. Othman and Ibrahim Mahmood Ibrahim, "A review of exploring recent advances in ant colony optimization: applications and improvements," International Journal of Scientific World, vol. 11, no. 1, pp. 114–122, Mar. 2025, doi: 10.14419/S0SJGQ84.

[39] S. Lipsa and R. Dash, "SLA-based task scheduling in cloud computing using randomized PSO algorithm," Jul. 2024.

[40] C. P. Shabariram and P. P. Ponnuswamy, "Task offloading in edge computing using integrated particle swarm optimization and genetic algorithm," Advances in Science and Technology Research Journal, vol. 19, no. 1, pp. 371–380, 2025, doi: 10.12913/22998624/195658.

[41] F. S. Prity, M. H. Gazi, and K. M. A. Uddin, "A review of task scheduling in cloud computing based on nature-inspired optimization algorithm," Cluster Comput, vol. 26, no. 5, pp. 3037–3067, Oct. 2023, doi: 10.1007/S10586-023-04090-Y.

[42] K. Sathupadi, "COMPARATIVE ANALYSIS OF HEURISTIC AND AI-BASED TASK SCHEDULING ALGORITHMS IN FOG COMPUTING: EVALUATING LATENCY, ENERGY EFFICIENCY, AND SCALABILITY IN DYNAMIC, HETEROGENEOUS ENVIRONMENTS," Quarterly Journal of Emerging Technologies and Innovations, vol. 5, no. 1, pp. 23–40, Jan. 2020, Accessed: Jun. 21, 2025. [Online]. Available: https://vectoral.org/index.php/QJETI/article/view/133

[43] D. Alsadie, "Advancements in heuristic task scheduling for IoT applications in fog-cloud computing: challenges and prospects," PeerJ Comput Sci, vol. 10, 2024, doi: 10.7717/PEERJ-CS.2128.

[44] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges," IEEE Communications Surveys and Tutorials, vol. 20, no. 1, pp. 416–464, Jan. 2018, doi: 10.1109/COMST.2017.2771153.

[45] C. Mouradian, S. Kianpisheh, M. Abu-Lebdeh, F. Ebrahimnezhad, N. T. Jahromi, and R. H. Glitho, "Application Component Placement in NFV-Based Hybrid Cloud/Fog Systems with Mobile Fog Nodes," IEEE Journal on Selected Areas in Communications, vol. 37, no. 5, pp. 1130–1143, May 2019, doi: 10.1109/JSAC.2019.2906790.

[46] N. M. Dankolo et al., "Optimizing resource allocation for IoT applications in the edge cloud continuum using hybrid metaheuristic algorithms," Sci Rep, vol. 15, no. 1, pp. 1–23, Dec. 2025, doi: 10.1038/S41598-025-97648-2;SUBJMETA=1042,117,258,639,705;KWRD=COMPUTATIONAL+SCIENCE,COMPUTER+SCIENCE,INFORMATION+TECHNOLOGY.

[47] G. Garai and B. B. Chaudhurii, "A novel hybrid genetic algorithm with Tabu search for optimizing multi-dimensional functions and point pattern recognition," Inf Sci (N Y), vol. 221, pp. 28–48, Feb. 2013, doi: 10.1016/J.INS.2012.09.012.

[48] Y. Zheng, Y. Chen, C. Tan, Y. Yang, C. Shu, and L. Chen, "Optimization model for vehicular network data queries in edge environments," Journal of Cloud Computing, vol. 13, no. 1, pp. 1–14, Dec. 2024, doi: 10.1186/S13677-024-00705-W/TABLES/2.

[49] Z. Nan, L. Wenjing, L. Zhu, L. Zhi, L. Yumin, and N. Nahar, "A new task scheduling scheme based on genetic algorithm for edge computing," Computers, Materials and Continua, vol. 71, no. 1, pp. 843–854, 2022, doi: 10.32604/CMC.2022.017504.

[50] S. Kaur and A. Verma, "An Efficient Approach to Genetic Algorithm for Task Scheduling in Cloud Computing Environment," International

Journal of Information Technology and Computer Science, vol. 4, no. 10, pp. 74–79, Sep. 2012, doi: 10.5815/IJITCS.2012.10.09.

[51] F. Xu, Z. Qin, L. Ning, and Z. Zhang, "Research on computing offloading strategy based on Genetic Ant Colony fusion algorithm," Simul Model Pract Theory, vol. 118, p. 102523, Jul. 2022, doi: 10.1016/J.SIMPAT.2022.102523.

[52] H. Nouhas, A. Belangour, and M. Nassar, "A Weighted Scoring Model Analysis of Cloud Storage Services: Comparing AWS, Azure, and Google Cloud Platform," International Journal of Engineering Trends and Technology, vol. 73, no. 6, pp. 350–370, Jun. 2025, doi: 10.14445/22315381/IJETT-V73I6P129.

[53] S. U. Mushtaq, S. Sheikh, and S. M. Idrees, "Enhanced priority based task scheduling with integrated fault tolerance in distributed systems," International Journal of Cognitive Computing in Engineering, vol. 6, pp. 152–169, Dec. 2025, doi: 10.1016/J.IJCCE.2024.12.006.

Optimization of Convolutional Neural Network Algorithm for Indonesian Sign Language Classification

Alvin Bintang Rebrastya[1], Sumarni Adi[2]*, Hanif Al Fatta[3],
Windha Mega Pradnya Dhuhita[4], Ika Nur Fajri[5], Muhammad Hanafi[6]
Department of Informatics, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia [1]
Faculty of Computer Science, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia [2, 3, 4, 5]
Magister of Informatics Engineering, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia [6]

***Abstract*—Sign language serves as a primary mode of communication for individuals who are deaf or speech impaired, using hand gestures to convey meaning visually. While it facilitates communication among the deaf community, it presents challenges for interaction with those who rely on spoken language. This study aims to recognize hand signs representing the letters A to Y (excluding J and Z) in the Indonesian Sign Language (SIBI) using image-based input. A custom dataset was collected through personal photo shoots and used to train a Convolutional Neural Network (CNN) implemented in Python using the TensorFlow library. The study also focuses on optimizing the CNN architecture to achieve high classification accuracy. Evaluation using a confusion matrix on the test data resulted in an overall accuracy of 87.1%, while real-time testing achieved an accuracy of 90.25%. The number of convolutional filters and dropout rates was adjusted to prevent underfitting and overfitting during model training.**

***Keywords*—Indonesian Sign Language; hand sign recognition; image classification; Convolutional Neural Network**

I. Introduction

According to the World Health Organization, more than 5% of the global population, approximately 430 million people, experience hearing loss greater than 35 decibels (dB) [1]. In Indonesia, around 7.03% of individuals with disabilities are categorized as deaf [2]. Deaf individuals use sign language to communicate, forming words and sentences through hand gestures. Sign language is a visual-gestural language that relies on movement and perception through vision [3]. It enables communication within the deaf community; however, interactions with people who rely on spoken language remain challenging.

In Indonesia, two main sign language models are used: the Indonesian Sign System (SIBI) and the Indonesian Sign Language (BISINDO). SIBI employs right-hand movements to represent alphabetical characters and is officially standardized by the Indonesian Ministry of Education and Culture [4]. BISINDO, in contrast, uses both hands and has developed organically within Indonesian society, resulting in regional variations. While BISINDO is widely used informally, SIBI remains the formal and standardized model.

Recent advancements in deep learning technology offer potential solutions for automating sign language recognition. One prominent algorithm in this domain is the Convolutional Neural Network (CNN), a type of deep learning model designed to process visual data [5]. Despite many studies applying CNNs for sign language recognition in various countries [6–11], there is limited research focusing on Indonesian Sign Language, particularly the SIBI model, using custom datasets collected in varied real-world conditions. This gap highlights the need for approaches tailored to the Indonesian context.

This study aims to address this gap by developing and optimizing a CNN-based model for classifying 24 SIBI hand signs (excluding J and Z). The model is trained on a custom image dataset collected via smartphone and webcam. The objectives of this research are to:

1) Design a CNN architecture suitable for classifying Indonesian SIBI signs.

2) Evaluate the impact of convolutional filter configurations and dropout rates on model performance.

3) Validate the model's accuracy through offline and real-time testing using confusion matrix analysis.

By providing a lightweight, efficient, and accurate CNN model trained on a custom dataset, this study contributes to improving accessibility and communication support tools for the Indonesian deaf community.

II. Related Works

Various kinds of previous studies have been proposed to recognize sign language. Sign languages in various countries have different signs, such as studies using Indian [6], Pakistani [7], Arabic [8], Bangla [9], Brazilian (Libras) [10], and Indonesian Sign Language with the BISINDO model [11].

In [6], A CNN-based Indian Sign Language (ISL) recognition system was developed to enable real-time hand gesture detection via a camera, achieving over 90% accuracy and was deployed on mobile devices using TensorFlow Lite and Flutter to enhance communication accessibility for the deaf-mute community. Research [7] Principal Component Analysis (PCA) and K-Nearest Neighbors (KNN) were applied, achieving 85% accuracy on test data and 80% when evaluated via webcam. Research [8] utilized the Faster R-CNN method.

*Corresponding author.

Using the ResNet-18 architecture yielded an accuracy of 93.4%, while VGG-16 achieved 93.2%. The work in [9] employed the YOLOv4 object detection model, obtaining an overall detection accuracy of 97.95%. In [10], a CNN-based approach with the VGGNet architecture was implemented. In [17], the study explored several input image sizes: 32×32 pixels (97.98% accuracy), 64×64 (99.42%), 128×128 (97.40%), and 256×256 (86.99%). Study [11] also adopted a CNN method, comparing multiple architectures. The modified version of AlexNet achieved an accuracy of 98.6%, while the modified VGG-16 architecture underperformed with only 3.8%. A novel architecture introduced in the same study, referred to as the C model, reached 98.3% accuracy.

Although various deep learning models have been applied in previous studies, such as Faster R-CNN [8], YOLOv4 [9], and modified architectures of AlexNet and VGG [10] [11], most of them require complex setups or large training resources. Furthermore, these models were developed for different language systems and may not generalize well to the Indonesian context. Compared to these methods, CNN offers a balanced trade-off between accuracy, computational efficiency, and ease of implementation.

This study focuses on a lightweight CNN model that is suitable for small-scale deployment with limited GPU resources. Our approach eliminates the need for extensive pre-training or large-scale datasets, making it more feasible for real-world applications in Indonesia.

III. METHODS

The research methodology in this study consists of several stages: data collection, data transformation, data splitting, CNN model training, prediction, and evaluation. An overview of these stages is illustrated in Fig. 1.

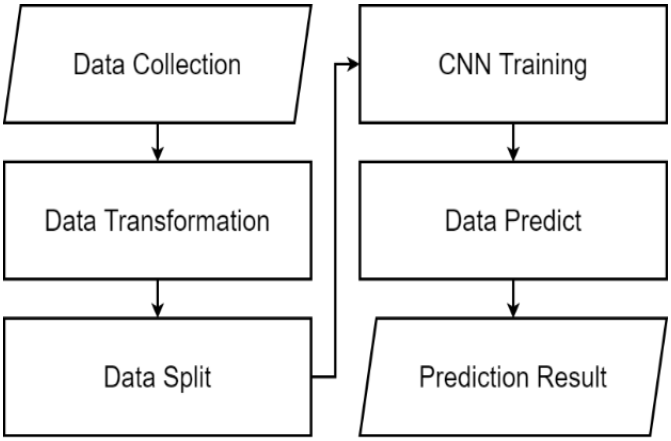


Fig. 1. Research stages.

The study began with the collection of sign language data. A total of 5,280 images were captured through self-recording using smartphones and webcam cameras. The next stage involved data transformation, which included resizing, grayscale conversion, data augmentation, and normalization.

Once transformed, the data were divided into two main sets: training and testing data. The core phase of the study was the training of a CNN model using the prepared dataset. Following training, the model was evaluated to assess its accuracy. Subsequently, testing was conducted using both the test dataset and newly captured images from a webcam to evaluate the model's ability to classify and predict sign language gestures in real time.

The following are some basic points that will be used in this research:

A. Data

The data used in this study consists of images representing Indonesian Sign Language (SIBI) hand gestures using the right hand. The dataset was collected independently [12], capturing variations in hand position, lighting, and background. The devices used for data collection included smartphone cameras and laptop webcams. The full dataset is accessible online [12].

B. Convolutional Neural Network

Convolutional Neural Networks (CNNs) are widely used for tasks such as speech recognition, image classification, and 3D object recognition. A typical CNN architecture includes two main components: a feature learning component and a fully connected layer. The feature learning section consists of convolutional layers, ReLU activation functions, and pooling layers. The fully connected layer is composed of neurons that process the extracted features to perform final classification. An overview of the general CNN architecture is presented in Fig. 2.

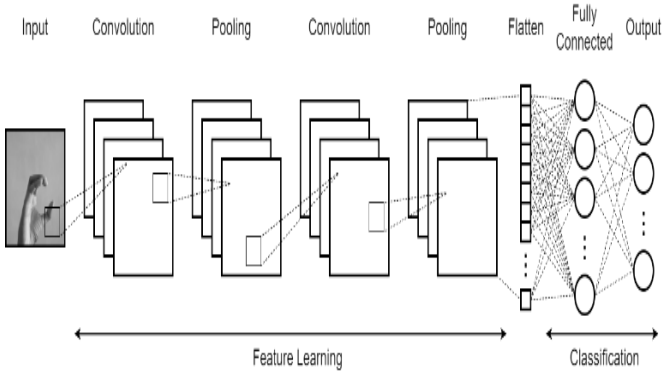


Fig. 2. Architecture of the CNN.

Convolution is the fundamental operation in CNNs, involving a matrix-based filtering process. This operation applies a filter (or kernel) matrix to an input image matrix to extract features, as defined in [13]:

$$\text{net}[i,j] = (x * w)[i,j] = \sum_m \sum_n x[m,n]\, w[i-m, j-n] \quad (1)$$

In this equation, w denotes the image matrix and x denotes the filter matrix. The convolution of these matrices results in a new feature map representing the output image. Indices i and j refer to the image matrix, while m and n refer to the kernel.

This study proposes a simple architecture consisting of the following layers:

- Convolutional layer 0,
- Max pooling layer 0,
- Convolutional layer 1,

- Dropout layer 0,
- Convolutional layer 2,
- Max pooling layer 1,
- Dropout layer1,
- Flatten layer
- Two fully connected layers.

This proposed architecture is illustrated in Fig. 3.

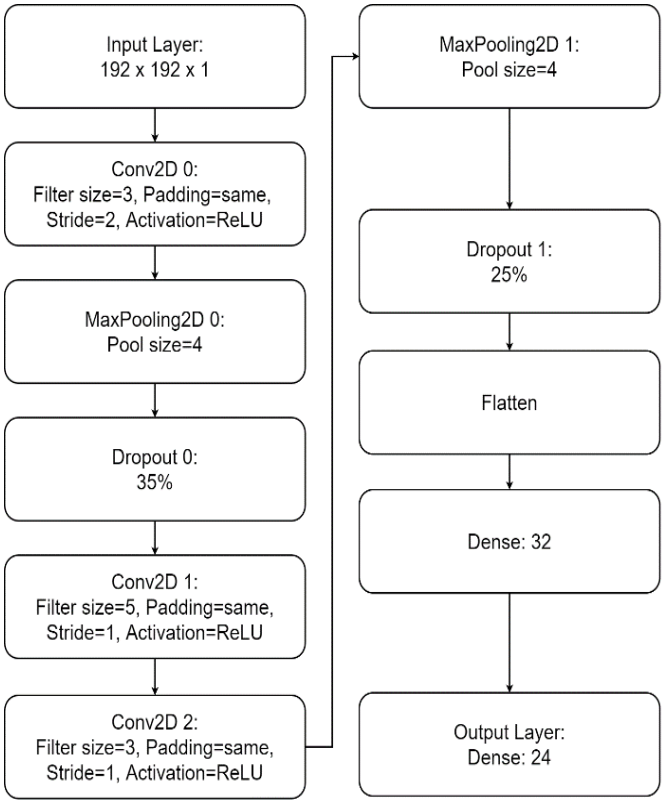


Fig. 3. Architecture model.

1) Baseline comparison and validation design: To validate the effectiveness of the proposed CNN architecture, we compared it with a basic CNN baseline model using fewer layers and no dropout. Both models were trained on the same dataset using identical preprocessing and training parameters. The baseline model consisted of a single convolutional layer followed by pooling and a fully connected layer.

Performance comparison between the proposed and baseline models is summarized in Table II. The proposed model achieved a higher accuracy (87.1%) compared to the baseline model (74.5%), and showed better generalization, as indicated by closer training and validation accuracy curves. Furthermore, F1-score, precision, and recall metrics improved by more than 10% on average per class.

This result confirms that the architectural optimizations—particularly in the number of convolutional filters and dropout rates—significantly improved classification performance and minimized overfitting.

C. Evaluation

The proposed CNN model was evaluated using a confusion matrix. This matrix produces four types of values: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These values were used to calculate performance metrics, including accuracy, precision, recall, and F1-score for each class. The confusion matrix serves as a basis for assessing the classification model's performance. Table I presents the layout of the confusion matrix [14].

TABLE I. CONFUSION MATRIX

Prediction	Actual Value	
	Positive	**Negative**
Positive	True Positive (TP)	False Positive (FP)
Negative	False Negative (TN)	True Negative (TN)

Definitions:

- True Positive (TP): Positive instances correctly classified.
- True Negative (TN): Negative instances correctly classified.
- False Positive (FP): Negative instances incorrectly classified as positive.
- False Negative (FN): Positive instances incorrectly classified as negative.

Based on these values, the following metrics were computed:

1) Overall Accuracy: Proportion of correct predictions out of total predictions [15][16]:

$$\text{Overall Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total Number of predictions}} \quad (2)$$

2) Per-class Accuracy: Accuracy calculated for each class individually (macro average), while overall accuracy is a micro average [15].

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \quad (3)$$

3) Precision: Measures the accuracy of positive predictions.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \quad (4)$$

4) Recall (Sensitivity): Measures the model's ability to correctly detect positive instances.

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \quad (5)$$

5) F1-score: The harmonic mean of precision and recall.

$$\text{F1} - \text{score} = 2 \text{ x } \frac{\text{precision x recall}}{\text{precision}+\text{recall}} \quad (6)$$

IV. RESULTS

A. Dataset Collection

The dataset used in this study is privately collected, comprising 5,280 images captured through self-recording [12].

The data consist of two types of images: 1) high-resolution RGB images (3000×3000 pixels) taken using a smartphone camera, and 2) lower-resolution grayscale images (250×250 pixels) taken using a webcam.

The dataset includes 24 hand gestures representing the SIBI alphabet (letters A to Y, excluding J and Z), with 220 images per class. An example of the dataset is shown in Fig. 4.



Fig. 4. Data sample.

B. Data Transformation

This stage involves transforming the data into a format suitable for training. The following techniques were applied:

1) Resize: All images were resized to 192×192 pixels and converted to grayscale to reduce computational load during training.

2) Grayscale: RGB images, composed of Red, Green, and Blue layers, were converted to grayscale using a weighted average of RGB channels. The grayscale transformation process is illustrated in Fig. 5 and Fig. 6.

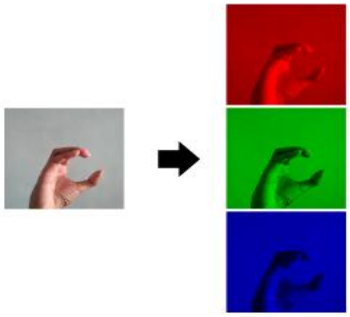


Fig. 5. RGB Data sample



Fig. 6. Grayscale image.

3) Data augmentation: Augmentation techniques were applied to increase dataset diversity. Thirty images per class were selected for transformation using random rotation (0.085), random zoom (0.1), and random translation (0.15). Images exceeding frame bounds were padded with black (zero constant). Sample augmentation results are shown in Fig. 7.

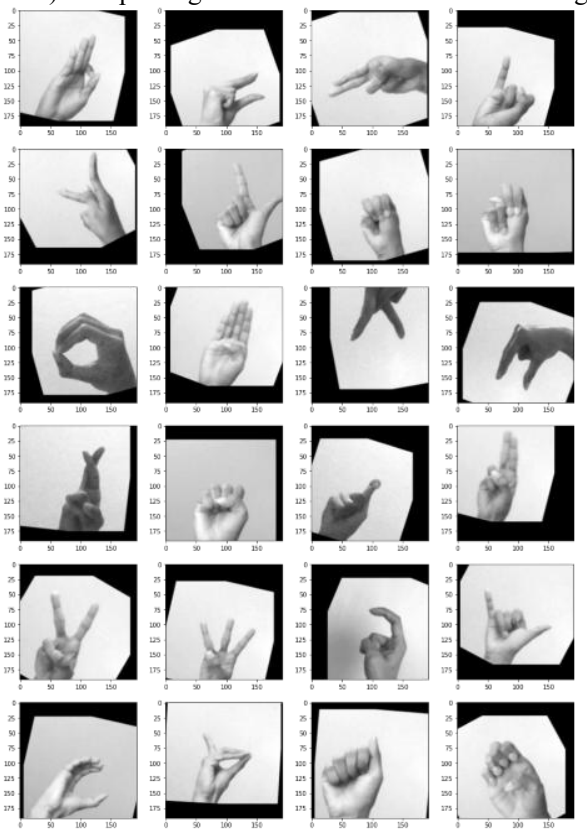


Fig. 7. Augmentation data sample.

C. Data Split

After augmentation, the dataset increased to 6,000 images (5,280 original + 720 augmented). The data were split as follows: 68% for training, 17% for validation, and 15% for testing.

D. Training Process

The CNN model was trained using a computer with an Intel Core i5-10210U CPU and NVIDIA GeForce MX330 GPU (2GB VRAM, GDDR5). A total of 720 images were used for testing.

Five different CNN configurations were evaluated by adjusting the number of convolution filters and dropout rates, while keeping filter size, stride, and padding constant. Table II lists the model architectures tested. The training used several parameters shown in Table III.

TABLE II. TRAINING PARAMETERS

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Loss Function	Categorical Cross-entropy
Epochs	80

TABLE III. MODEL ARCHITECTURE TESTING

No.	Parameter				
	Conv0	Dropout0	Conv1	Conv2	Dropout1
1	48	-	96	192	-
2	24	-	48	96	-
3	24	0,35	48	96	0,25
4	48	0,35	96	192	0,25
5	24	0,85	48	48	0,75

E. Testing Model

Testing the model from the CNN algorithm that has been made, then testing using a confusion matrix by testing the image from the test data, as many as 15% or 900 data units. This test was carried out five times to determine the level of accuracy, duration of training, accuracy graph, loss graph, and whether the two graphs were overfit or underfit. The parameters that are changed are the number of convolution filters and dropouts, while the filter size, stride, and padding in the filter are not changed. Table IV is the result of a comparison of the training that has been done.

F. Real-Time Testing

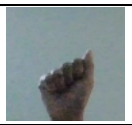
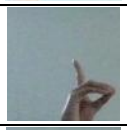
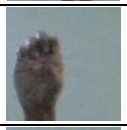
The architecture used for real-time testing corresponds to configuration number 3 in Table IV, which previously demonstrated the best performance. The testing was conducted using the right hand to perform SIBI (Sistem Isyarat Bahasa Indonesia) sign language gestures within the detection area of a standard laptop webcam. The system was implemented as a desktop application.

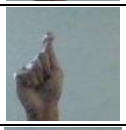
The testing results, including accuracy per class, are listed in Table V. The model achieved an average accuracy of 90.25%. However, lower accuracy was observed for the letters M and N, which can be attributed to their visual similarity, making them more difficult to distinguish during classification.

TABLE IV. TRAINING COMPARISON RESULTS

No.	Accuracy	Training Duration
1	82,4	10m 39,3s
2	80,0	6m 8,4s
3	87,1	6m 20,4s
4	90,3	11m 6,2s
5	46,6	5m 44,9s

TABLE V. REAL-TIME TEST RESULTS

Input	Class	Result (%)
	A	89
	B	97
	C	98
	D	96
	E	91
	F	89
	G	96
	H	100
	I	88
	K	97

Input	Class	Result (%)
	L	96
	M	70
	N	63
	O	93
	P	84
	Q	84
	R	89
	S	91
	T	90
	U	94
	V	91
	W	93
	X	96
	Y	91

V. Discussion

The results of the training, along with the corresponding graphs and confusion matrix shown in Table III, are presented as follows:

1) First test: In this test, the model configuration corresponds to Test 1 in Table IV. The training and validation accuracy graphs, along with the confusion matrix, are shown in Fig. 8 and Fig. 9. The graph indicates signs of overfitting, as the validation accuracy plateaus after the 20th epoch and exhibits a substantial gap compared to the training accuracy. Additionally, the loss graph shows a noticeable increase starting from the 45th epoch, further confirming the presence of overfitting.

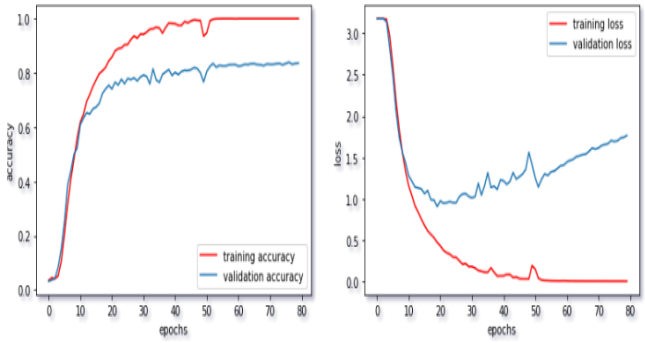


Fig. 8. Graph of the first test result.

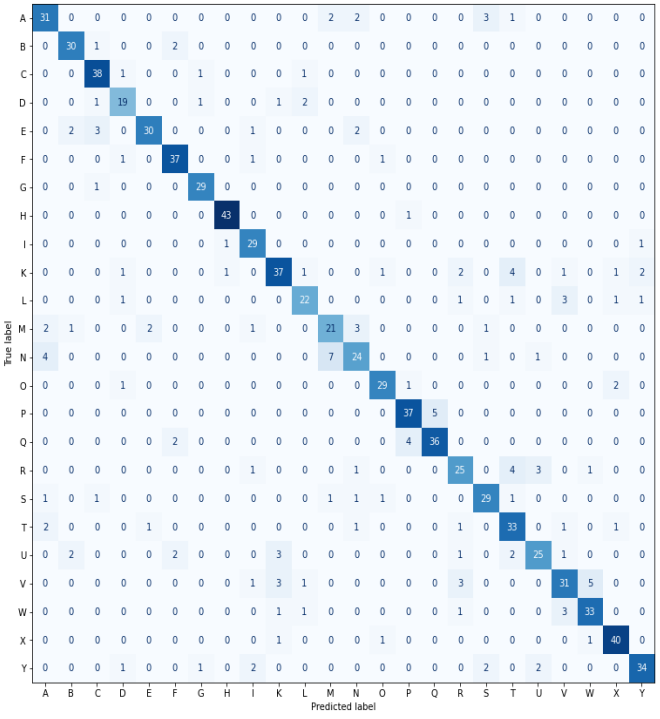


Fig. 9. Confusion matrix for the first test.

2) Second test: This test uses the configuration listed as Test 2 in Table IV. The training and validation graph, as well as the confusion matrix are presented in Fig. 10 and Fig. 11. According to [18], overfitting can occur when the neural

network fails to improve its performance during training. In this case, signs of overfitting appear after the 20th epoch, where validation accuracy stagnates and deviates significantly from training accuracy. The loss curve also rises steadily without decreasing as training progresses.

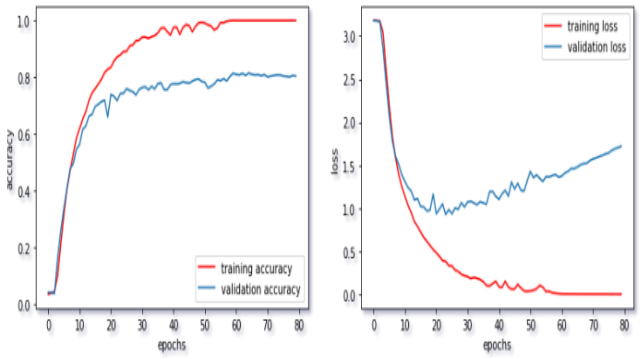


Fig. 10. Graph of the second test result.

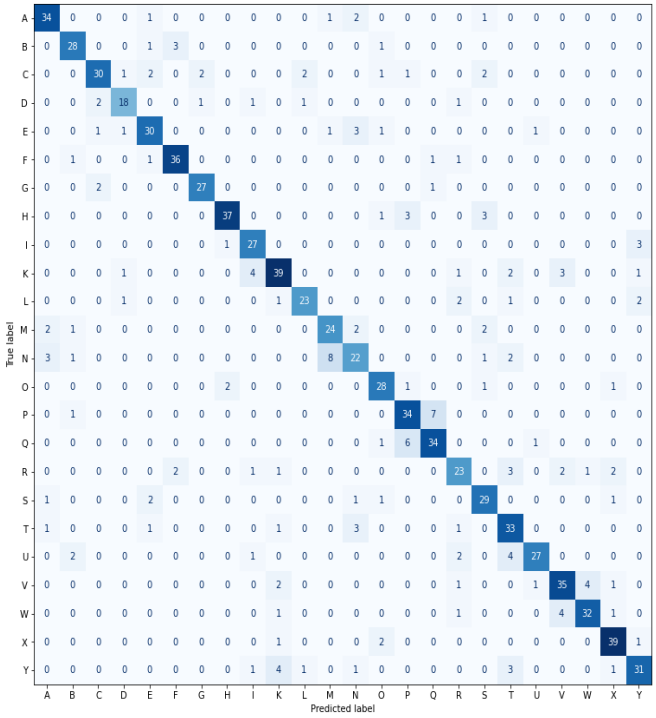


Fig. 11. Confusion matrix for the second test.

3) Third test: The configuration for this test is listed as Test 3 in Table IV. The training and validation graphs, along with the confusion matrix, are shown in Fig. 12 and Fig. 13. In this test, validation accuracy closely follows training accuracy, and the validation loss also aligns well with training loss throughout the training process. This indicates that the model avoids both underfitting and overfitting, resulting in high and stable accuracy.

4) Fourth test: The model configuration in this test is listed as Test 4 in Table IV. The corresponding graphs and confusion matrix are presented in Fig. 14 and Fig. 15. As noted in [9], convolutional layers with a large number of filters tend to require longer training times. This configuration yields the highest accuracy among the tests; however, slight overfitting is observed after the 25th epoch, where the validation accuracy shows minimal improvement.

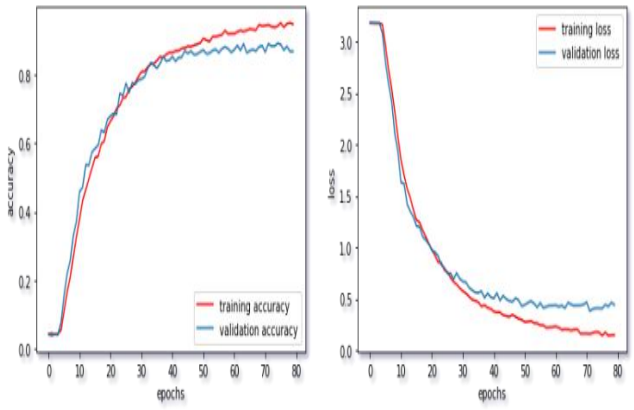


Fig. 12. Graph of the third test result.

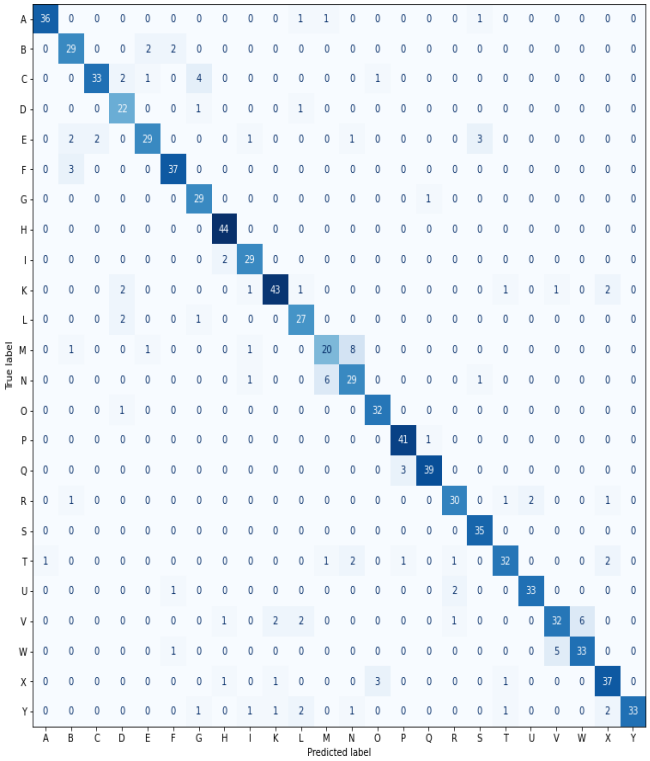


Fig. 13. Confusion matrix for the third test.

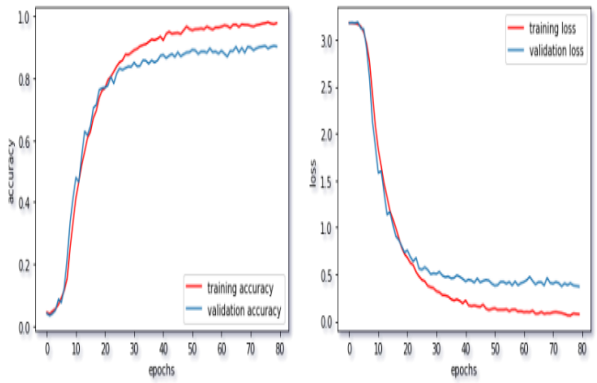


Fig. 14. Graph of the fourth test result.

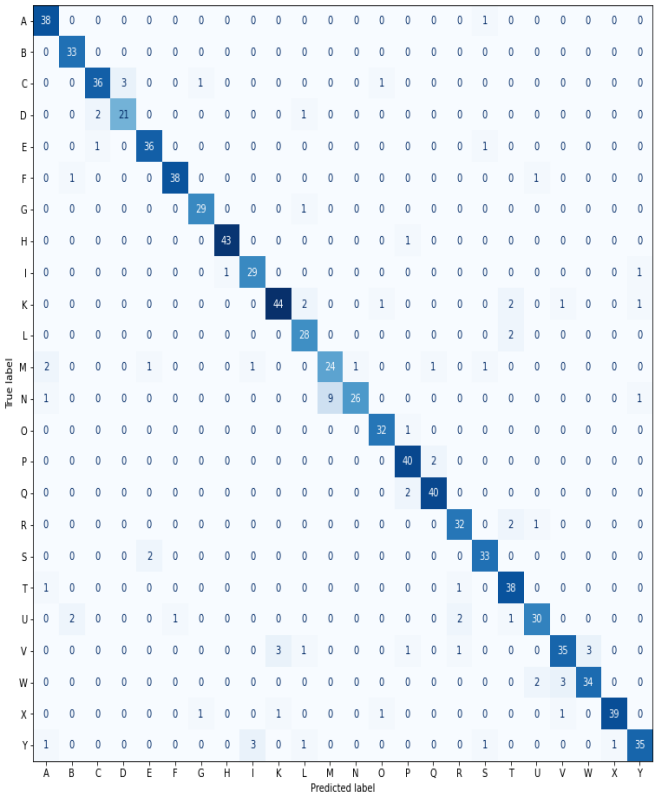


Fig. 15. Confusion matrix for the fourth test.

5) Fifth Test: This test uses the configuration listed as Test 5 in Table IV. The corresponding training graphs and confusion matrix are shown in Fig. 16 and Fig. 17. This model employs the lowest number of filters among all configurations. As stated in [13], network performance can be influenced by the depth of the architecture. In this test, the model achieves the lowest accuracy and the highest loss. The graph indicates underfitting, as the validation accuracy exceeds the training accuracy, suggesting the model is not learning effectively.

Among the five model architectures tested in Table IV, Test 3 yields the best performance, achieving high accuracy without signs of overfitting or underfitting. The final confusion matrix used to calculate per-class accuracy, precision, recall, and F1-score is presented in Fig. 18. While this figure demonstrates high overall accuracy, the precision, recall, and F1-score values are relatively lower, indicating potential room for improvement in class-level predictions.

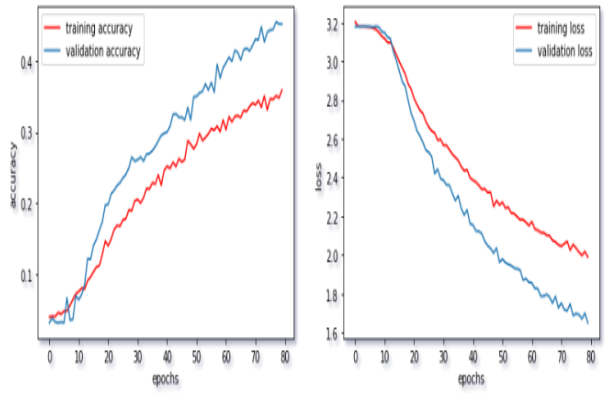


Fig. 16. Graph of the fifth test result.

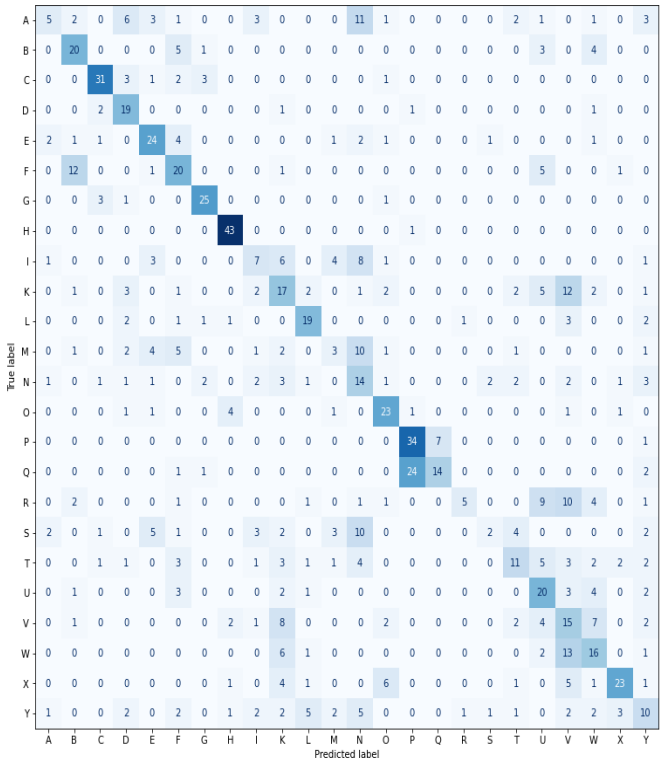


Fig. 17. Confusion matrix for the fifth test.

Compared to the baseline CNN model without architectural optimization, the proposed model demonstrates a significant improvement in both accuracy and robustness. While the baseline model produced acceptable results for simple gesture recognition, it suffered from overfitting and lower generalization on real-time webcam input.

Our optimized architecture achieved performance comparable to or exceeding that of related studies [9][10], despite being trained on a smaller, personalized dataset. This highlights that even lightweight CNN architectures can deliver strong performance when properly tuned and applied to targeted data.

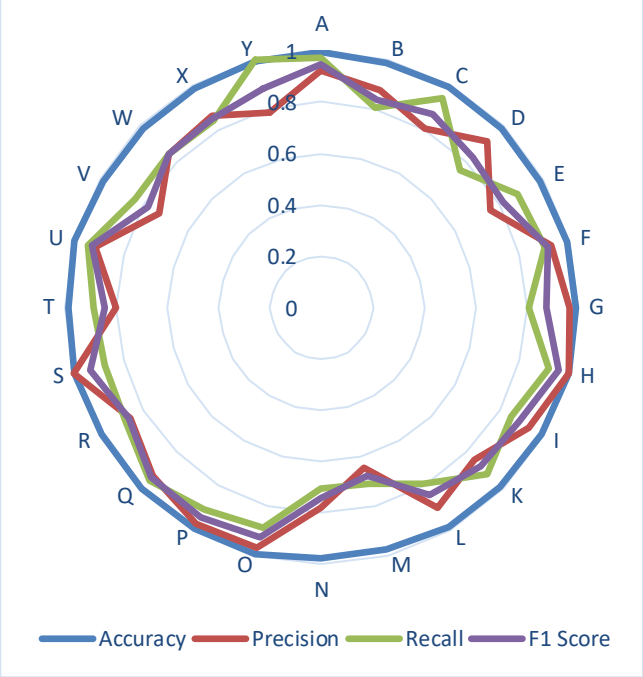


Fig. 18. Confusion matrix result.

Despite the promising results, this study has several limitations. The dataset used was relatively small and collected in controlled conditions, which may not fully capture variations in lighting, background, and hand positioning encountered in real-world environments. Furthermore, the scope of recognition was restricted to 24 SIBI alphabet signs, excluding letters J and Z due to their dynamic motion-based representation. These limitations may affect the model's generalizability when applied to more diverse or continuous sign language inputs.

Nevertheless, the findings demonstrate that lightweight CNN architectures can be effectively optimized to deliver robust performance even with modest datasets. This highlights the potential for deploying such models in practical applications, such as educational tools, real-time communication aids, and mobile platforms designed to bridge communication gaps for the deaf community.

Future research could focus on expanding the dataset to include more participants, environmental variations, and dynamic gestures such as J and Z. Integrating the model with real-time mobile or web-based applications would also provide practical validation of its effectiveness in everyday scenarios. Additionally, exploring advanced architectures, such as transfer learning or hybrid CNN–RNN models, may further improve recognition accuracy and support continuous sign language translation.

VI. Conclusion

This study demonstrates that optimizing the number of convolutional filters and dropout rates significantly improves the performance of CNN-based models for Indonesian Sign Language (SIBI) recognition. The best configuration achieved 87.1% accuracy on the test set and 90.25% accuracy in real-time scenarios, with no indication of overfitting or underfitting.

The results indicate that lightweight CNN architectures, when properly tuned, can perform effectively even in resource-constrained environments. Compared to heavier architectures like Faster R-CNN, YOLOv4, AlexNet, or VGG reported in other studies, the proposed model offers advantages in terms of computational efficiency and speed. However, a direct empirical comparison on the same dataset remains to be conducted to fully validate claims of superiority.

Currently, the system relies on cropped hand images, limiting its flexibility in real-world scenarios. Automatic hand detection through object detection methods has not yet been implemented, representing an important avenue for future improvement.

Future work will focus on extending the model to recognize dynamic gestures, such as the letters J and Z, integrating object detection techniques to automatically identify relevant hand regions, and deploying the system as a mobile-based application for real-time sign language interpretation. Incorporating these enhancements will improve robustness and usability, making the system more practical for inclusive communication tools.

Acknowledgment

This research was funded through a collaboration between the Department of Information Systems and the Department of Research and Community Services at Universitas Amikom Yogyakarta, Indonesia.

References

[1] World Health Organization, "Deafness and Hearing Loss," 2021. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss (accessed Jul. 09, 2022).

[2] A. Harpini, "Disabilitas Rungu," Pusat Data dan Informasi Kementerian Kesehatan Republik Indonesia, 2019. https://pusdatin.kemkes.go.id/resources/download/pusdatin/infodatin/infodatin-tunarungu-2019.pdf (accessed Jul. 10, 2022).

[3] S. T. Isma, "Meneliti Bahasa Isyarat dalam Perspektif Variasi Bahasa," 2018. [Online]. Available: http://kbi.kemdikbud.go.id/kbi_back/file/dokumen_makalah/dokumen_makalah_1540468871.pdf.

[4] "Kamus SIBI," Kementerian Pendidikan dan Kebudayaan dan Lembaga Penelitian dan Pengembangan Sistem Isyarat Bahasa Indonesia, 2020. https://pmpk.kemdikbud.go.id/sibi/ (accessed Jul. 14, 2022).

[5] TensorFlow Team, "Distributed training with TensorFlow," TensorFlow, 2024. [Online]. Available: https://www.tensorflow.org/guide/distributed_training.

[6] A. Deshmukh, "Real-Time Indian Sign Language Recognition Using CNNs for Communication Accessibility," International Journal of Multidisciplinary Research and Analysis, vol. 7, no. 9, pp. 4447-4453, 2024, doi: 10.47191/ijmra/v7-i09-41.

[7] M. S. Arshad Malik, N. Kousar, T. Abdullah, M. Ahmed, F. Rasheed, and M. Awais, "Pakistan sign language detection using PCA and KNN," Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 4, pp. 78–81, 2018, doi: 10.14569/IJACSA.2018.090414.

[8] R. A. Alawwad, O. Bchir, and M. M. Ben Ismail, "Arabic Sign Language Recognition using Faster R-CNN," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 3, pp. 692–700, 2021, doi: 10.14569/IJACSA.2021.0120380.

[9] D. Talukder and F. Jahara, "Real-Time Bangla Sign Language Detection with Sentence and Speech Generation," ICCIT 2020 - 23rd Int. Conf. Comput. Inf. Technol. Proc., no. April, 2020, doi: 10.1109/ICCIT51783.2020.9392693.

[10] J. Rocha, J. Lensk, T. Ferreira, and M. Ferreira, "Towards a Tool to Translate Brazilian Sign Language (Libras) to Brazilian Portuguese and Improve Communication with Deaf," Proc. IEEE Symp. Vis. Lang.

Human-Centric Comput. VL/HCC, vol. 2020-Augus, no. July, 2020, doi: 10.1109/VL/HCC50065.2020.9127257.

[11] S. Dwijayanti, Hermawati, S. I. Taqiyyah, H. Hikmarika, and B. Y. Suprapto, "Indonesia Sign Language Recognition using Convolutional Neural Network," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 10, pp. 415–422, 2021, doi: 10.14569/IJACSA.2021.0121046.

[12] A. B. Rebrastya, "Indonesian Sign Language (SIBI) Dataset," 2022. https://www.kaggle.com/datasets/alvinbintang/sibi-dataset.

[13] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," Proc. 2017 Int. Conf. Eng. Technol. ICET 2017, vol. 2018-Janua, no. April 2018, pp. 1–6, 2018, doi: 10.1109/ICEngTechnol.2017.8308186.

[14] V. Inacio, F. Carvalho, R. Rodrigues, dan A. Ferreira, "Statistical evaluation of medical tests," arXiv, 2020. [Online]. Available: https://arxiv.org/abs/2007.07687.

[15] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 2, pp. 612–619, 2020, doi: 10.14569/ijacsa.2020.0110277.

[16] Z. Mahmood, L. Jamel, D. A. Salem, and I. Ashraf, "Improving learning from the complex multi-class imbalanced and overlapped data by mapping into higher dimension using SVM++," Scientific Reports, vol. 15, no. 31245, 2025. [Online]. Available: https://www.nature.com/articles/s41598-025-13929-w.

[17] T. Al-Shehari, M. Kadrie, M. N. Al-Mhiqani, T. Alfakih, H. Alsalman, M. Uddin, S. S. Ullah, and A. Dandoush, "Comparative evaluation of data imbalance addressing techniques for CNN-based insider threat detection," Scientific Reports, vol. 14, no. 24715, 2024. [Online]. Available: https://www.nature.com/articles/s41598-024-73510-9.

[18] H. Li, G. K. Rajbahadur, D. Lin, C.-P. Bezemer, Z. Ming, and J. Jiang, "Keeping deep learning models in check: A history-based approach to mitigate overfitting," arXiv, 2024. [Online]. Available: https://arxiv.org/abs/2401.10359.

Control System of Ocean Wave Simulator Using PID-Salp Swarm Algorithm

Affiani Machmudah[1], Juchen Li[2], Mahmud Iwan Solihin[3], Chiong Meng Choung[4],
Wibowo Harso Nugroho[5], Ahmad Syafiul Mujahid[6], Sahlan[7], Abdul Ghofur[8]
Research Center for Satellite Technology, National Research and Innovation Agency (BRIN), Bogor, Indonesia[1]
Faculty of Engineering-Technology and Built Environment, UCSI University, Kuala Lumpur, Malaysia[2, 3, 4]
School of Intelligent Manufacturing, Anhui Wenda University of Information Engineering, Hefei, China[2]
Research Center for Hydrodynamics Technology, National Research and Innovation Agency (BRIN), Surabaya, Indonesia[5, 6, 7, 8]

***Abstract*—This paper presents a control system optimization of an ocean wave simulator using a meta-heuristic optimization. The proposed control system involves finding leg length trajectories by an Inverse Kinematics (IK) to be used as references for a Proportional-Integral-Derivative (PID). PID gains are tuned using a Salp Swarm Algorithm (SSA) with a Root Mean Square Error (RMSE) of leg position errors as a performance index. A Stewart platform dynamic is modeled in Simscape Multibody, integrated with a trajectory generator, an IK module, and a control system diagram block in Simulink models. The Simulink model of Stewart platform dynamics is then called in the optimization procedure by writing the programming code in MATLAB. Results show that the SSA outperforms other meta-heuristic methods, namely a Genetic Algorithm (GA) and a Particle Swarm Optimization (PSO), achieving the lowest fitness value, 16.8% and 8.7% lower than GA and PSO, respectively. Moreover, the SSA avoids boundary-trapping issues encountered by the PSO, which is stuck at its upper bound. The SSA has successfully enhanced the simplified version of the PID control system, where the scenario of the simplified PID-SSA scenario achieves better tracking error performance than the full PID-SSA configuration. The proposed approach contributes to the advancement of marine simulation technologies, supporting innovation in ocean engineering and sustainable maritime applications.**

Keywords*—Marine simulation technologies, Stewart platform; ocean wave; control system; meta-heuristic optimization; Salp Swarm Algorithm

I. Introduction

The ocean is a vital part of the Earth and plays a significant role in the global ecosystem. Since approximately 70% of the Earth's surface is covered by ocean, conducting research in the maritime field is essential. One key challenge in this area is managing and understanding ocean waves. However, testing newly developed marine technologies directly in the real ocean is often difficult and costly, especially due to the complexity of transporting and deploying such systems. As a result, modeling and simulation have become crucial areas of maritime research. Simulation technologies can replicate real ocean conditions, allowing researchers to observe and analyze the performance of the developed systems effectively.

Generally, there are two types of ocean wave simulators: simulation platforms and ocean engineering basins [1]. A simulation platform applies wave models to control a motion platform, while an ocean engineering basin uses known ocean wave models as input for controlling a wave-making system. One of the most widely used simulation platforms for ocean wave simulation is the Stewart platform. The Stewart platform has been extensively utilized in the development and testing of ocean technology applications ranging from ship motion simulation [1-5], harbor crane robotics technologies [6-8], to ocean wave compensators [9-13].

The Stewart platform is a parallel manipulator that has superiorities in accuracy, rigidity, and capability in carrying heavy loads that to the series manipulator. Controlling the parallel manipulator is difficult due to its complex, coupled nonlinear dynamics, demanding advanced methods for reliable performance. As the Stewart platform is expected to execute complex motions with stability, its control systems remain a critical area of research. Silva *et al*. [14] employed the industrial controller on the Stewart platform, addressing kinematics and motion automation. The IK considered two scenarios: point-to-point cycloidal trajectories and oceanic wave motion. Vu *et al*. [15] presented a distributed control technique based on finite-time distributed sliding mode control. Each leg of the Stewart platform was considered as an autonomous agent. The dynamic of the Stewart platform was modeled using a Lagrangian-based approach. The trajectory tracking control by the Stewart platform using a PD+ controller was presented by Tamir *et al*. [16]. The dynamic of the Stewart platform was modelled using the Euler-Lagrange method. The motion was simulated using MATLAB/Simulink and validated using ADAM software. Lin *et al*. [17] presented drag-free control of the Stewart platform for satellite ground simulation. Inverse Kinematics (IK) and dynamics of the Stewart platform based on Newton-Euler were employed. Two degree-of-freedom drag-free controller was proposed using an algorithm of H_∞ loop shaping. An adaptive control of the pneumatic actuator of the sixth degree-of-freedom Stewart platform was presented by Andrievsky *et al*. [18]. Pneumatic valve groups and a spool valve were involved. Control process, discretization, and pneumatic drive load changes were considered.

Artificial Intelligent methods such as meta-heuristic optimization and an Artificial Neural Network (ANN) have been employed in the control system of the Stewart platform. Barghandan *et al*. [19] proposed a sliding mode control based on the Genetic Algorithm (GA) considering uncertainties and

external disturbances. An adaptation law based on a barrier function had removed the requirement of the external disturbance upper bounds. Tamir *et al*. [20] presented an extended Proportional-Derivative (PD) sliding mode controller for the Stewart platform trajectory tracking. Gain tuning was performed using modified grey wolf optimization with integral absolute error as a cost function. Validation was performed using the ADAM software. Sumnu *et al*. [21] presented the Proportional-Integral-Derivative (PID) control system of the Stewart manipulator, where the linear motor transfer function is derived and modeled in Simulink block diagram. The effect of the inertia of the lower legs on the dynamic performance was studied. Velasco *et al*. [22] presented the ANN to improve the accuracy of the Stewart platform when it holds a camera. The ANN model was designed based on the kinematics and dynamics analysis. Based on the inverse ANN model and additional integral control, the control system was then developed. Du *et al*. [13] applied the Stewart platform as a marine stabilized platform and proposed an extended state observer-based sliding mode control (ESO-SMC) to enhance the stability and anti-disturbance of the hydraulic actuator on the system platform. Kinematic and dynamic models were analyzed first, and the Marine Predator Algorithm (MPA) was applied to optimize the ESO-SMC. Simulation was conducted in MATLAB in Simulink using models of an engineering ship and a marine stabilized platform, which was placed on top of the ship model. An *et al*. [23] designed a Fracture Reduction Robot (FRR) based on the Stewart platform as an orthopedic surgery robot. A sparrow search algorithm was applied to tune the controller gains of the FRR. The IK was used, and the FRR dynamics were modeled in Simulink. The mechanism of fracture reduction was proposed using computer torque control.

The paper is organized as follows: Section II presents the kinematics and dynamics of a 6-DOF Stewart platform-based ocean wave simulator. Section III describes the motion simulator system. The trajectories generation, the control system optimization, and the Salp Swarm Algorithm (SSA) are presented in Section IV. Results and discussions are presented in Sections V and VI, respectively. Conclusions are presented in Section VII.

II. Kinematics and Dynamics

Kinematics and dynamics analysis are necessary for controlling the Stewart platform-based ocean wave simulator. For the dynamics analysis, this paper models the Stewart platform using Simscape Multibody to be optimized by the SSA via MATLAB and Simulink environments. MATLAB and Simulink have been used to model complex dynamical systems, including the Stewart platform to analyze the proposed methods before implementing them in the real system.

A. Inverse Kinematics

There are two kinds of kinematics problems for Stewart platform motion, namely Inverse kinematics (IK) and Forward Kinematics (FK). The FK is the problem of finding the positions and orientations of payload platform when leg lengths are defined while the IK is the problem of finding the leg lengths of Stewart platform when positions and orientations of payload platform are known. This paper considers tracking the prescribed paths, which are the ocean wave model so that using the 6-DOF Stewart platform, converting the tracking paths to the leg lengths trajectories is necessary to be done via an IK computation. Fig. 1 illustrates the coordinates assigned for the Stewart platform. There is a base platform coordinate {B} located at the center of the base table and a payload platform coordinate {P} located at the center of the payload table.

A closed-form IK solution can be obtained using geometrical and algebraic analysis as follows [24]:

$$l_i^2 = x^2 + y^2 + z^2 + r_P^2 + r_B^2 + 2\left(r_{11}P_{ix} + r_{12}P_{iy}\right)(x - b_{ix}) + 2\left(r_{21}P_{ix} + r_{22}P_{iy}\right)\left(y - b_{iy}\right) + 2\left(r_{31}P_{ix} + r_{32}P_{iy}\right)z - 2\left(xb_{ix} - yb_{iy}\right) \quad (1)$$

The above equation needs the orientation matrix. The orientation of the Roll-Pitch-Yaw angle is expressed as follows:

$$R_{RPY} = \begin{bmatrix} c\alpha\, c\beta & c\alpha\, s\beta\, s\gamma - s\alpha\, c\gamma & c\alpha\, s\beta\, c\gamma - s\alpha\, s\gamma \\ s\alpha\, c\beta & s\alpha\, s\beta\, s\gamma - c\alpha\, c\gamma & s\alpha\, s\beta\, c\gamma - c\alpha\, s\gamma \\ -s\beta & c\beta\, s\gamma & c\beta\, c\gamma \end{bmatrix} \quad (2)$$

where, R_{RPY}, α, β, γ, s, and c are the orientation matrix, roll angle, pitch angle, yaw angle, sinus, and cosinus, respectively.

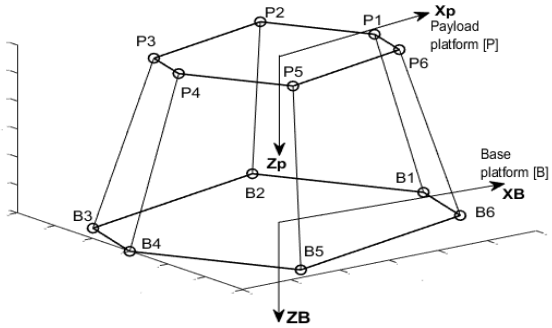


Fig. 1. Frame coordinate of the Stewart platform.

B. Dynamics

Dynamics of the Stewart platform can be expressed in the following form [25]:

$$F = M(q)\ddot{q} + N(q, \dot{q}) + G(q) \quad (3)$$

where, $M(q)$ is the inertia matrix, $N(q, \dot{q})$ is components of Coriolis and centrifugal force, $G(q)$ is the components of gravity force.

The inverse dynamics is the problem of finding the force/torque for known positions, velocities, and accelerations for defined positions, velocities, and acceleration vectors. This paper employs Simscape Multibody for solving the forward dynamics of the Stewart platform in the control system. Detail of the Simscape block diagram is shown in Fig. 2.

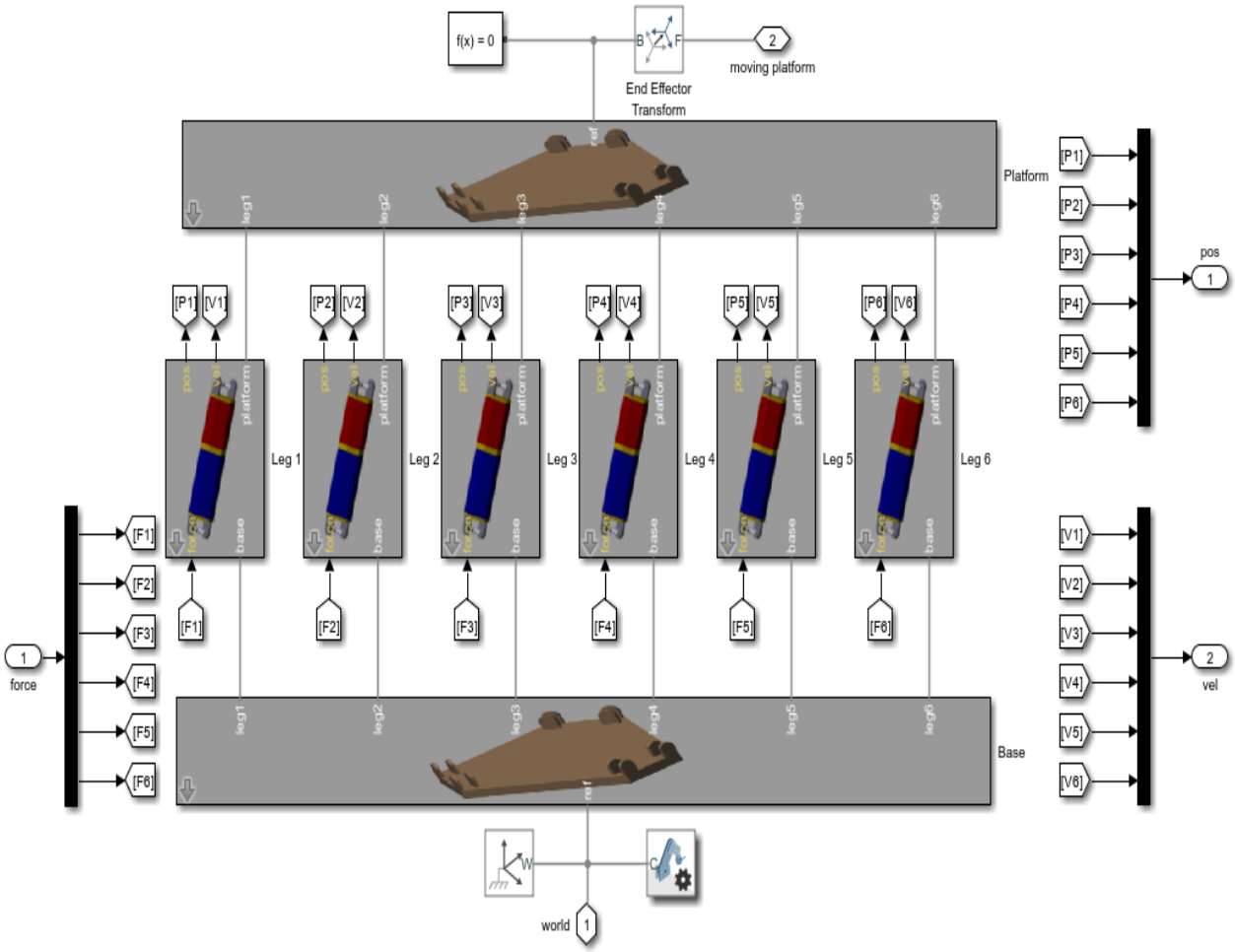


Fig. 2. Dynamic modeling of the Stewart platform using Simulink Multibody.

III. Motion Simulator System

Fig. 3 illustrates the ocean wave simulator system. The traced path is generated by the trajectory generator. An IK module processes the data from the trajectory generator to obtain the requested leg positions, which become the input for the control system of the Stewart platform.

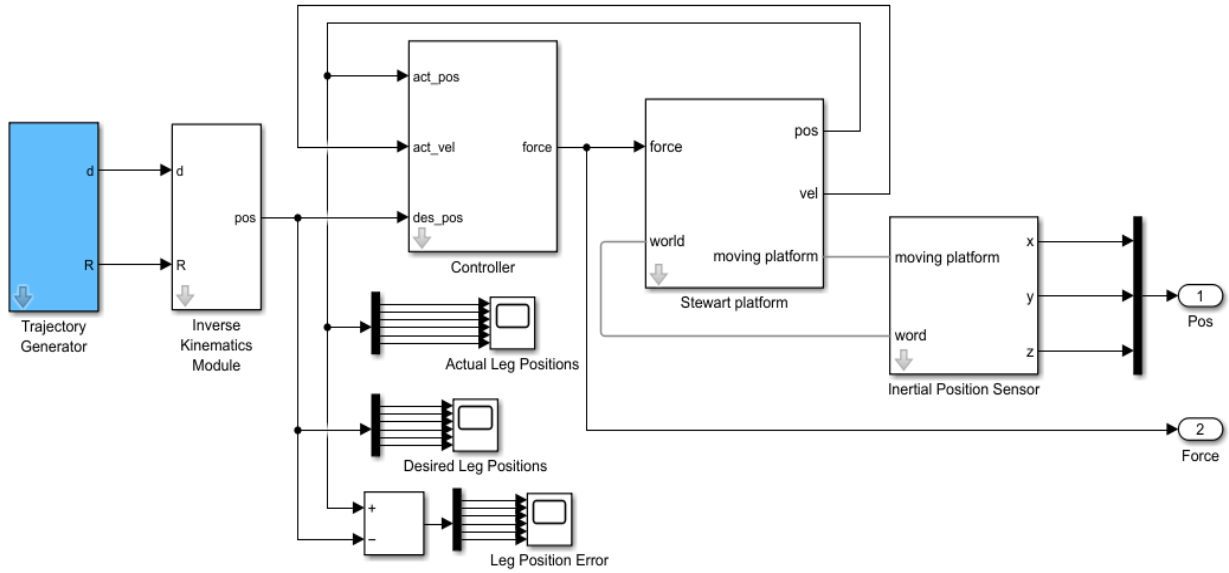


Fig. 3. Ocean wave simulator system.

A. Trajectory Generations

Ocean wave models used in this paper is presented in Table I. They are the modification of ocean wave models presented in [1]. This modification is necessary because the parameter of the Stewart platform is different and the Stewart platform has the workspace which depends on the parameter of the Stewart platform.

TABLE I. OCEAN WAVE MODELS [1]

Motion forms	Model (cm)
Lateral	$8\sin(0.25\pi t)$
Transverse	$8\sin(0.25\pi t)$
Lifting	$8\sin(0.25\pi t)$
Rolling	$\frac{10\pi}{180}\sin(0.25\pi t)$
Pitching	$\frac{8\pi}{180}\sin(0.25\pi t)$

IV. CONTROL SYSTEM OPTIMIZATIONS

Fig. 5 illustrates the algorithm of the control system optimization. Ocean wave models need to be converted to leg length trajectories via the IK. These leg length trajectories become the reference trajectories that need to be tracked by the control system of the motion simulator. This paper uses the PID-based control system to be optimized by the SSA. A comprehensive computation is necessary through integration of MATLAB software, Simulink, and Simscape Multibody. The responses of the Stewart platform are obtained using the Simulink to examine the Simscape Multibody modeling of Stewart platform.

Fig. 4 shows the block diagram of the ocean wave model represented in Table I. The Stewart platform uses in this paper has parameter as presented in Tabel II.

Finding the optimal gains by meta-heuristic optimization are performed by writing the code in m-file, MATLAB. Using the IK presented in the previous section, the trajectories of leg lengths can be obtained. These leg lengths trajectories should be within the Δl_{max} boundary. In the case the leg length trajectories are outside the boundary, the initial condition, which is the initial distance between the fixed base and moving base H, possibly needs to be recalculated. The Δl_{max} can be obtained by the geometric analysis of the Stewart platform.

A. PID Control System

Leg lengths trajectories from the trajectories generator become the reference signal of the control system. This paper uses the conventional PID control system optimized by the SSA. The detail of the simplified PID control system as the Simulink block diagram is illustrated in Fig. 6.

TABLE II. STEWART PLATFORM PARAMETER

Parameters	Values
Radius of moving platform (hexagon)	18 cm
Radius of fixed platform (hexagon)	28 cm
Cylinder radius of leg	3 cm
Maximum upper leg length	10 cm
Maximum lower leg length	30 cm
Initial distance between the fixed base and moving platform, H	13.5 cm

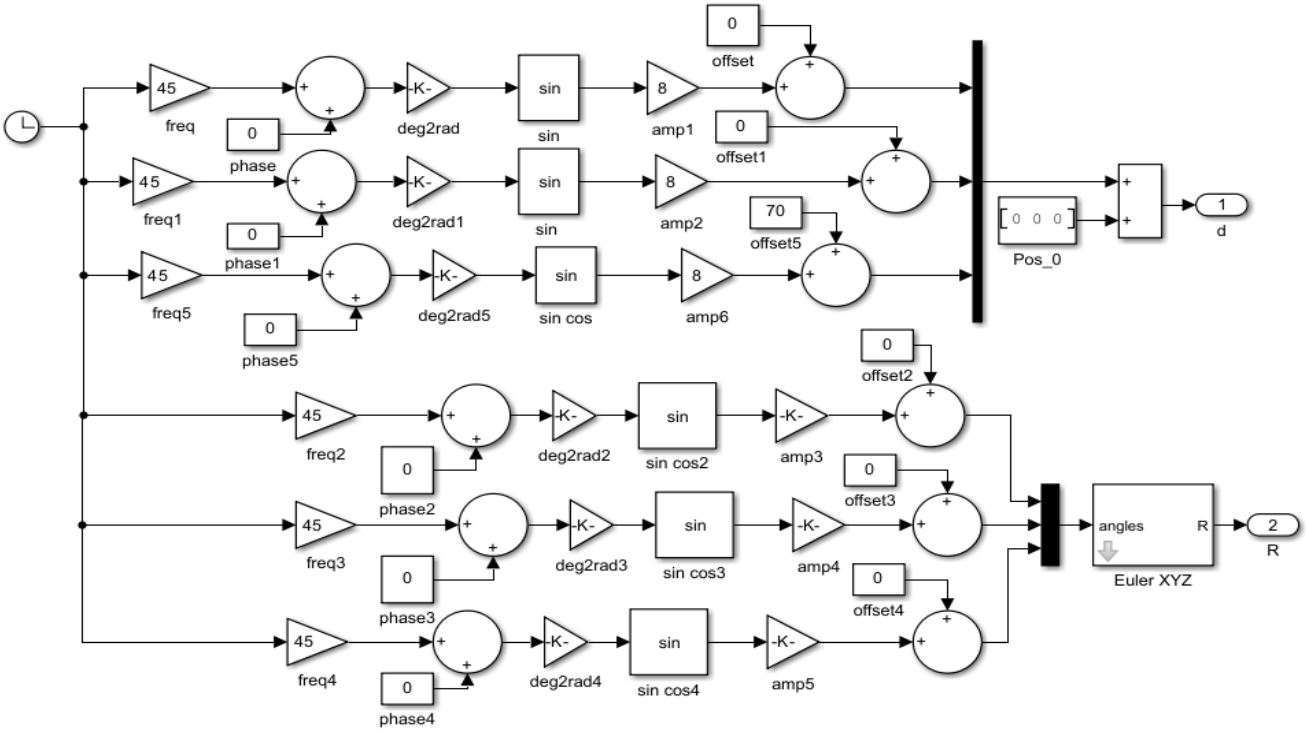


Fig. 4. Diagram block of trajectory generation of ocean wave motion.

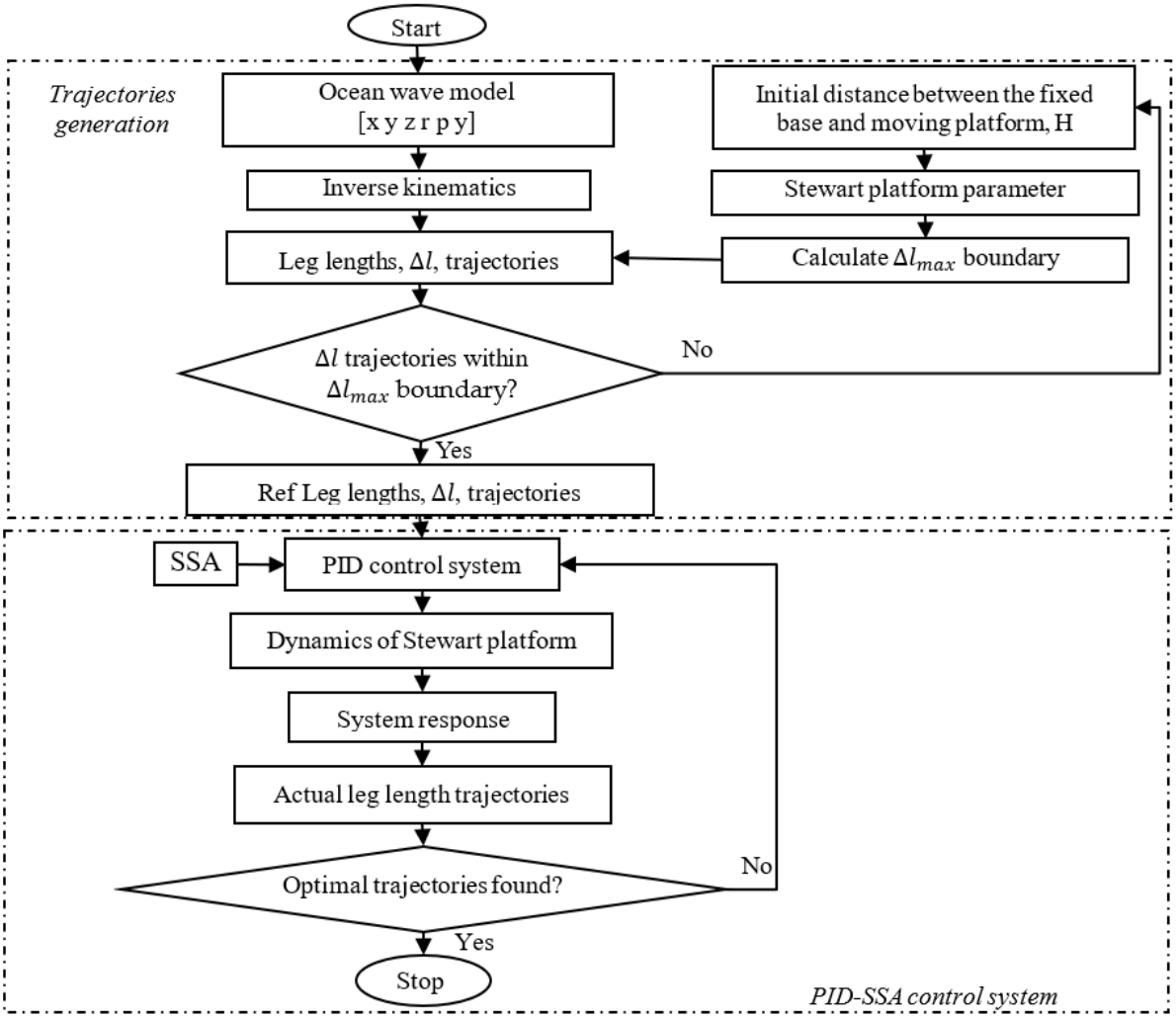


Fig. 5. Control system algorithm of the Stewart platform.

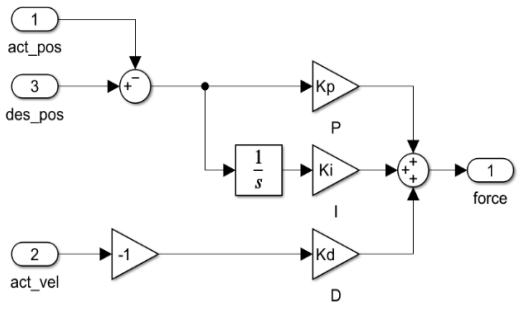


Fig. 6. Simplified conventional PID control system block diagram.

The simplified conventional PID control law is employed as follows:

$$F = K_p e + K_i \varepsilon - K_d \dot{\Delta l} \tag{4}$$

$$e = \left(\Delta l_{ref} - \Delta l\right) ; \dot{\varepsilon} = e$$

where, F, e, Δl, Δl_{ref}, $\dot{\Delta l}$, K_p, K_i, and K_d are force feedback control, error leg length, actual leg length, desired leg length, actual leg velocities or first derivative of Δl, proportional gain, integral gain, and derivative gain, respectively.

B. Salp Swarm Algorithm

The PID gains are optimized by the SSA with the objective optimization is to minimize the Root Mean Square Error (RMSE) as the cost function. The SSA is relatively new meta-heuristic optimization proposed in 2017 by Mirjalili *et al.* [26]. The SSA is inspired by the salps swarming behaviour when they are navigating and foraging in deep oceans.

The populations consists two groups, namely leader and followers. The leader is the salp at the chain front. The leader position is updated as follows:

$$x_j^1 = \begin{cases} F_j + c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right) & c_3 \geq 0 \\ F_j - c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right) & c_3 < 0 \end{cases} \tag{5}$$

where, x_j^1, F_j, ub_j, and lb_j, are the leader position in jth dimension, the food source in jth dimension, upper bound in jth dimension, lower bound in jth dimension, respectively. Parameters c_1, c_2, and c_3 are random numbers.

Above leader position is updated based on the food source. For exploration and exploitation, the following parameter is defined:

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \quad (6)$$

where, l and L is the current iteration and the maximum iteration number, respectively. The c_2, and c_3 parameters are generated randomly in interval [0, 1].

The follower position is updated as follows:

$$x_j^i = \frac{1}{2}\left(x_j^i - x_j^{i-1}\right) \quad (7)$$

where, $i \geq 2$ and x_j^i is the ith follower salp position in jth dimension.

V. Results

Besides the SSA, other meta-heuristic optimizations, which are the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), are also executed as comparison methods. The GA, PSO, and SSA use 10 individuals in the population evaluated for 50 iterations. The optimization parameters are searched within the search area [100, 10000] for K_p gain and [2, 1000] for the K_i and K_d gains. The result of Δl_{max} is 30 cm for the Stewart platform used in this paper (Table II). The computation of the gain optimization is performed within 2.5 seconds and divides the motion into 200 sampling times.

Fig. 7 shows the results of the leg length trajectories using the IK for the value of H = 13.5 cm. It can be observed that the leg lengths trajectories are within their workspace, where all trajectories are within Δl_{max} boundary.

A. Trial-and-Error Gains

It is generally known that tuning the PID gain is a challenging computational problem, while using a trial-and-error approach is a tedious task. The fitness value using this gain is 6.642 for the trial-and-error gain, [K_p K_i K_d] = [300 100 100]. Fig. 8 shows the system response for the trial-and-error gains. The difference between the desired trajectories and the actual trajectories is not small, so tuning the gains is necessary to obtain the efficient motion.

B. Optimal Gains

Results of the optimal gains by the GA, PSO, and SSA are presented in Table III. The computation of the RMSE as the fitness function is performed for 2.5 second running time. It shows that the SSA algorithm is outperformed the GA and PSO where the SSA has the lowest fitness value. It can also be observed that the SSA does not trap to the upper bound value and lower bound value.

The SSA has ability to explore and exploit the searching area through parameter c_1 in Eq. (6). On the contrary, the PSO has been trapped to the upper bound value for the K_p (K_p = 10000). Fig. 9 shows the fitness value evolution during 50 numbers iterations. The fitness value of the SSA is 16.8% and 8.7% lower than the GA and the PSO, respectively.

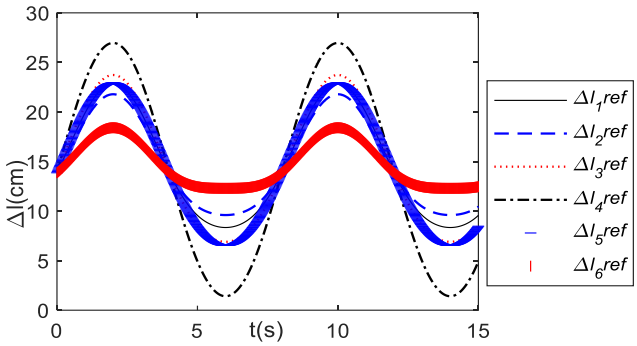


Fig. 7. Leg lengths trajectories of the ocean wave model within Δl_{max} boundary.

TABLE III. Results of GA, PSO, SSA

Methods	Fitness value	Best [K_p K_i K_d]
GA	0.0781	[9661.2 992.4 20]
PSO	0.072	[10000 6190 20]
SSA	*0.066*	[7472.4 747.24 5]

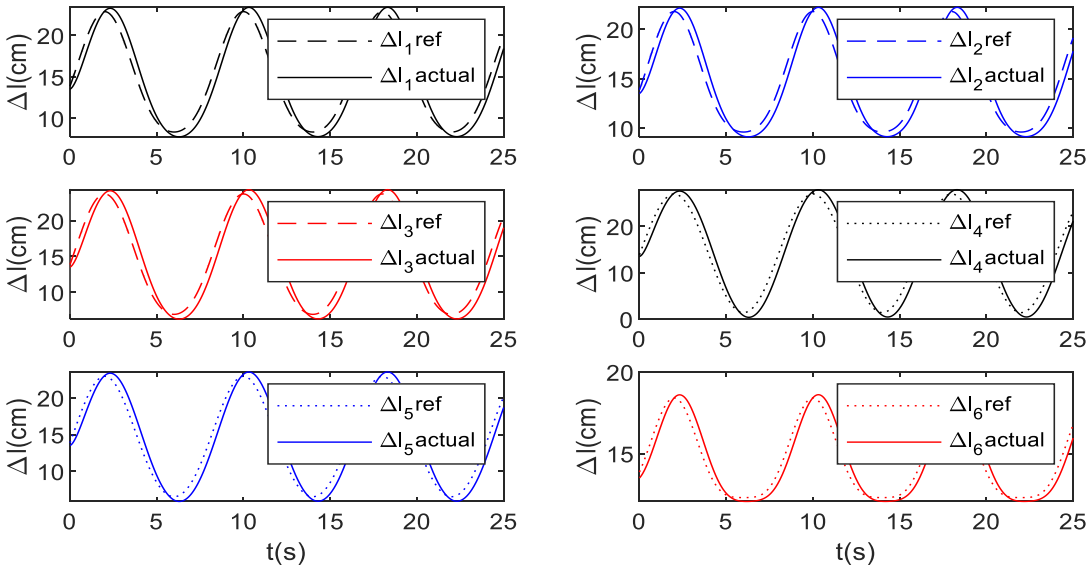


Fig. 8. System response for the trial and error approach [K_p K_i K_d] = [300 100 100].

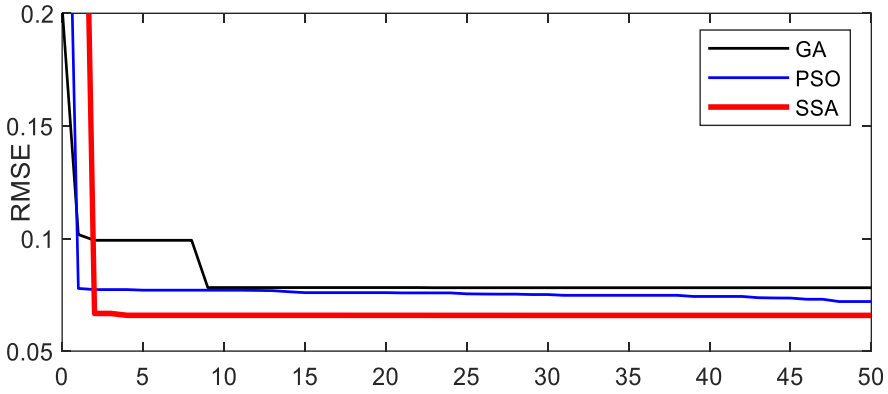


Fig. 9. Fitness value of GA, PSO, and SSA.

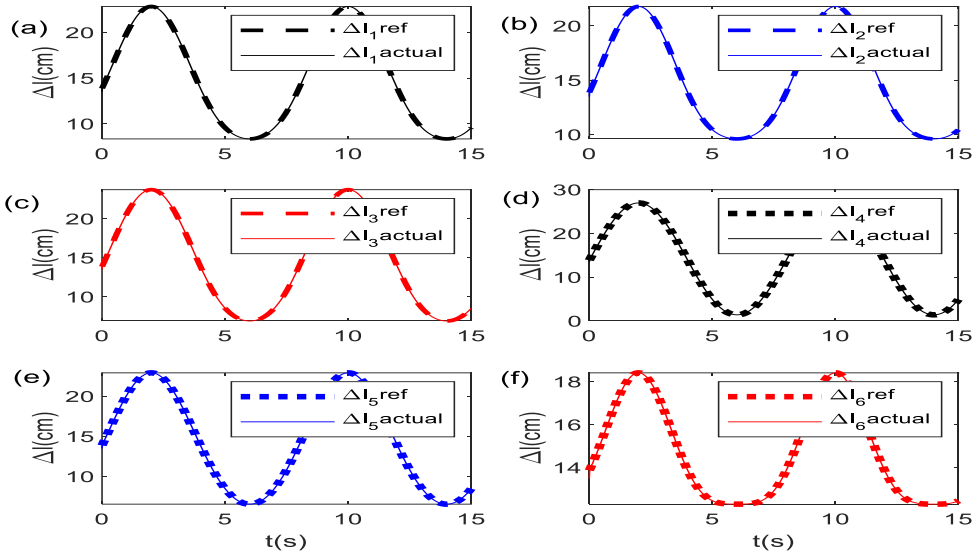


Fig. 10. System response for the SSA method, [K_v K_p K_i] = [7472.4 747.24 5].

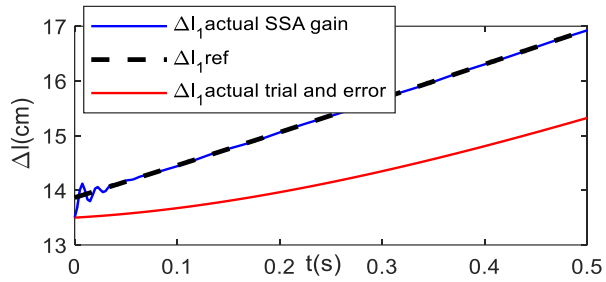
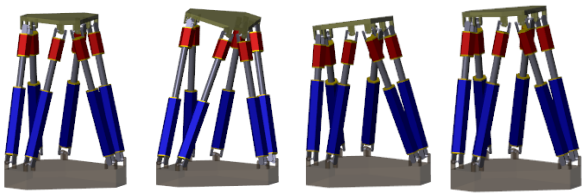


Fig. 11. Zoom in of system response by SSA and trial-and-error gain.



(a) (b) (c) (d)

Fig. 12. Motion frames of the ocean wave simulator K=[7472.4 747.24 5] (a) 0 second (b) 2.5 second (c) 5 second (d) 7.5 second.

Fig. 10 shows the system response of the SSA gains. As compare with the respond from the trial-and-error gain, it can be observed that the SSA has significantly improved the trial-and-error approach. Fig. 11 shows detail of comparison between the results of the SSA gains and the trial and error gains for fist leg length trajectories. Fig. 12 shows the motion frame of the Stewart platform during tracking the ocean wave model.

C. PID with Velocity Reference

The previous result was obtained from the simplified version of the PID control system without feedforward of the velocity setpoint. This section investigates the performance of full PID control system when the velocity reference is included as an input to the controller.

The full version of the PID control law, in which the velocity reference is used as the control input, is expressed as follows:

$$F = K_p e + K_i \varepsilon - K_d \dot{e} \tag{8}$$

$$e = \dot{\Delta l}_{ref} - \dot{\Delta l}$$

The detail of the block the Simulink block diagram is illustrated in Fig. 13.

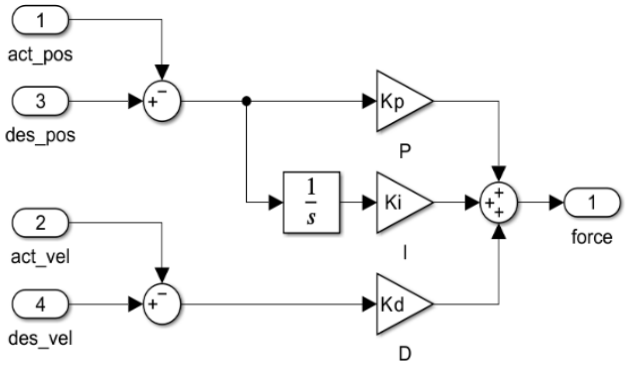


Fig. 13. Full version of PID control system block diagram.

Table IV presents the results of the trial-and-error and SSA gains for the PID control system with velocity reference input. By accommodating the velocity reference in the simplified conventional PID control, using the same trial-and-error gain K = [300 100 100], the fitness value has significantly improved to 0.7450. However, this value is still far from the SSA gain, where the fitness achieved to 0.066 (Table III).

VI. Discussions

The SSA has significantly improved the RMSE of the ocean wave simulator's motion. Fig. 14 presents the detailed system response for the trial-and-error gains of the full version PID control system. However, when the SSA gains from the previous section (see Table III) are applied with the velocity reference included, the resulting fitness value is 0.1052—worse than that of the simplified PID control system.

The performance of the full version of the PID control system is further investigated by tuning the gains using the SSA. Using the same computation parameters as in the simplified PID, the optimization result is presented in Table IV. The SSA gain for the PID control is K = [8148.97 814.9 b37.64] with the fitness value is 0.099443. Based on the fitness value result, the simplified PID-SSA has better performance than that of the PID-SSA.

TABLE IV. Results of SSA for Full Version of the PID System

Methods	Fitness value	$[K_p\ K_i\ K_d]$
Trial-and-error	0.7450	[300 100 100]
Gain from previous section	0.1052	[7472.4 747.24 5]
SSA	0.099443	[8148.97 814.9 37.64]

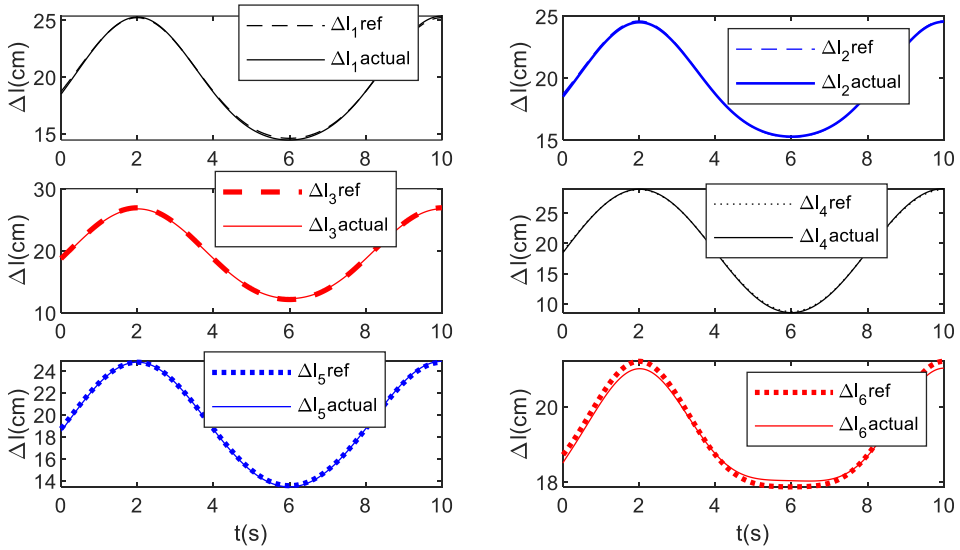


Fig. 14. System response of PID with velocity reference for the trial and error approach, $[K_p\ K_i\ K_d]$ = [300 100 100].

Employing the SSA in the PID control system of the ocean wave simulator has yielded promising results. Adding the velocity reference input into the PID improves the simulator's response for the trial-and-error gain, as reflected in the fitness value results. However, when the SSA is applied to correct the required force, the simplified PID (without the velocity reference input) outperformed the SSA-PID. The main challenge in reducing the position tracking error likely stems from the highly nonlinear dynamics of the 6-DOF Stewart platform. Standard PID control systems often require additional compensation to handle such nonlinearities effectively. Therefore, for future work, incorporating advanced nonlinear compensation methods, such as computed-force control, can be investigated to further enhance the system respond of the PID control system.

VII. Conclusions

The control system optimization of the Stewart-platform-based ocean wave simulator using the PID-SSA method has been presented. The system solved the IK to obtain leg length trajectories, used as reference inputs for the PID controller. By employing Simscape Multibody to handle the forward dynamics, IK module, and PID control, the SSA was applied to tune the PID gains. Results showed that the SSA not only success to identify the optimal gain, but also significantly enhance the performance of the simplified PID controller as

compare to the full version of the PID controller. The promising performance of SSA in improving simplified PID control suggests strong potential for its application to ocean wave compensators in ships and ocean floating platforms. Future work should address nonlinear compensation in simulator dynamics and extend the proposed method to wave-induced motion control on offshore platforms.

ACKNOWLEDGMENT

The research in this article was funded by the Professional Leader Training Program of Anhui Provincial Department of Education (No. DTR2024059). This research was supported by Riset dan Inovasi untuk Indonesia Maju (RIIM) Batch 4 by Lembaga Pengelola Dana Pendidikan (LPDP) (Grant No. 37/II.7/HK/2023).

REFERENCES

[1] W. Chuan, D. Huafeng, and H. Lei, “A dynamic ocean wave simulator based on six-degrees of freedom parallel platform,” Proc Inst Mech Eng C J Mech Eng Sci, vol. 232, no. 20, pp. 3722–3732, Oct. 2018, doi: 10.1177/0954406217739647.

[2] Z. Zhou, B. Zhang, J. Chang, Q. Dong, and H. Wang, “Dynamic response analysis of the vessel under the action of motion compensation equipment,” Ocean Engineering, vol. 315, Jan. 2025, doi: 10.1016/j.oceaneng.2024.119885.

[3] Z. Li, H. Du, J. Si, Z. Wang, and W. Xiong, “Rigid-flexible coupling dynamic modeling and validation of a helicopter rescue simulator based on an inverted Stewart platform,” Ocean Engineering, vol. 319, Mar. 2025, doi: 10.1016/j.oceaneng.2024.120164.

[4] T. Ono, R. Eto, J. Yamakawa, and H. Murakami, “Analysis and control of a Stewart platform as base motion compensators - Part I: Kinematics using moving frames,” Nonlinear Dyn, vol. 107, no. 1, pp. 51–76, Jan. 2022, doi: 10.1007/s11071-021-06767-8.

[5] T. Ono, R. Eto, J. Yamakawa, and H. Murakami, “Analysis and control of a Stewart platform as base motion compensators—part II: dynamics,” Nonlinear Dyn, vol. 106, no. 4, pp. 3161–3182, Dec. 2021, doi: 10.1007/s11071-021-06749-w.

[6] L. Liang, Z. Le, S. Zhang, and J. Li, “Modeling and controller design of an active motion compensated gangway based on inverse dynamics in joint space,” Ocean Engineering, vol. 197, Feb. 2020, doi: 10.1016/j.oceaneng.2019.106864.

[7] A. Niu, S. Wang, Y. Sun, J. Qiu, W. Qiu, and H. Chen, “Dynamic modeling and analysis of a novel offshore gangway with 3UPU/UP-RRP series-parallel hybrid structure,” Ocean Engineering, vol. 266, Dec. 2022, doi: 10.1016/j.oceaneng.2022.113122.

[8] B. Lu, Y. Fang, J. Lin, Y. Hao, and H. Cao, “Nonlinear antiswing control for offshore boom cranes subject to ship roll and heave disturbances,” Autom Constr, vol. 131, Nov. 2021, doi: 10.1016/j.autcon.2021.103843.

[9] Y. Cai, S. Zheng, W. Liu, Z. Qu, and J. Han, “Model analysis and modified control method of ship-mounted Stewart platforms for wave compensation,” IEEE Access, vol. 9, pp. 4505–4517, 2021, doi: 10.1109/ACCESS.2020.3047063.

[10] Q. Zhang, B. Tan, B. P. Gu, and X. Hu, “Docking ship heave compensation system for loading operations based on a DDPG and PID hybrid control method using a judge network,” Ocean Engineering, vol. 305, Aug. 2024, doi: 10.1016/j.oceaneng.2024.117727.

[11] Y. Wang, Y. Wei, W. Gao, T. Ma, and Y. Han, “Ocean Wave Active Compensation Analysis for Redundant Hybrid Boarding System: A Multi-Task Motion Planning Method,” J Mar Sci Eng, vol. 11, no. 4, Apr. 2023, doi: 10.3390/jmse11040708.

[12] W. Chen et al., “An ADRC-based triple-loop control strategy of ship-mounted Stewart platform for six-DOF wave compensation,” Mech Mach Theory, vol. 184, Jun. 2023, doi: 10.1016/j.mechmachtheory.2023.105289.

[13] Z. Du, X. Chen, Q. Zhang, and Y. Yang, “An extended state observer-based sliding mode control method for hydraulic servo system of marine stabilized platforms,” Ocean Engineering, vol. 279, Jul. 2023, doi: 10.1016/j.oceaneng.2023.114386.

[14] D. Silva, J. Garrido, and E. Riveiro, “Stewart Platform Motion Control Automation with Industrial Resources to Perform Cycloidal and Oceanic Wave Trajectories,” Machines, vol. 10, no. 8, Aug. 2022, doi: 10.3390/machines10080711.

[15] D. C. Vu, T. L. Nguyen, and D. H. Nguyen, “A novel approach of Consensus-based Finite-time Distributed Sliding Mode Control for Stewart platform manipulators motion tracking,” Results in Engineering, vol. 25, Mar. 2025, doi: 10.1016/j.rineng.2024.103872.

[16] T. S. Tamir, G. Xiong, Y. Tian and G. Xiong, "Passivity Based Control Of Stewart Platform For Trajectory Tracking," 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China, 2019, pp. 988-993, doi: 10.1109/ICIEA.2019.8833935

[17] J. Lin, J. Lian, G. Zhao, H. Li, and C. Xu, “A 2-DOF drag-free control ground simulation system based on Stewart platform,” ISA Trans, vol. 146, pp. 528–540, Mar. 2024, doi: 10.1016/j.isatra.2024.01.010.

[18] B. Andrievsky, N. V. Kuznetsov, E. V. Kudryashova, O. A. Kuznetsova, and I. Zaitceva, “Signal-parametric discrete-time adaptive controller for pneumatically actuated Stewart platform,” Control Eng Pract, vol. 138, Sep. 2023, doi: 10.1016/j.conengprac.2023.105616.

[19] M. Barghandan, A. A. Pirmohamadi, S. Mobayen, and A. Fekih, “Optimal adaptive barrier-function super-twisting nonlinear global sliding mode scheme for trajectory tracking of parallel robots,” Heliyon, vol. 9, no. 2, Feb. 2023, doi: 10.1016/j.heliyon.2023.e13378.

[20] T. S. Tamir et al., “Design and Optimization of a Control Framework for Robot Assisted Additive Manufacturing Based on the Stewart Platform,” Int J Control Autom Syst, vol. 20, no. 3, pp. 968–982, Mar. 2022, doi: 10.1007/s12555-021-0058-4.

[21] A. Şumnu, İ. H. Güzelbey, and M. V. Çakir, “Simulation and PID control of a Stewart platform with linear motor,” Journal of Mechanical Science and Technology, vol. 31, no. 1, pp. 345–356, Jan. 2017, doi: 10.1007/s12206-016-1238-7.

[22] J. Velasco, Ó. Barambones, I. Calvo, P. Venegas, and C. M. Napole, “Validation of a Stewart platform inspection system with an artificial neural network controller,” Precis Eng, vol. 74, pp. 369–381, Mar. 2022, doi: 10.1016/j.precisioneng.2022.01.002.

[23] B. An et al., “Optimization of fracture reduction robot controller based on improved sparrow algorithm,” Biomimetic Intelligence and Robotics, vol. 3, no. 4, Dec. 2023, doi: 10.1016/j.birob.2023.100120.

[24] C. C. Nguyen, Z. . -L. Zhou, S. S. Antrazi and C. E. Campbell, "Efficient computation of forward kinematics and Jacobian matrix of a Stewart platform-based manipulator," IEEE Proceedings of the SOUTHEASTCON '91, Williamsburg, VA, USA, 1991, pp. 869-874 vol.2, doi: 10.1109/SECON.1991.147884

[25] K. Harib and K. Srinivasan, “Kinematic and dynamic analysis of Stewart platform-based machine tool structures,” Robotica, vol. 21, no. 5, pp. 541–554, Sep. 2003, doi: 10.1017/S0263574703005046.

[26] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, “Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems,” Advances in Engineering Software, vol. 114, pp. 163–191, Dec. 2017, doi: 10.1016/j.advengsoft.2017.07.002.

Enhanced IoT Security Using Machine Learning Technology

Rawan Yousef Bukhowah, Alanoud Khaled Bu Dookhi, Mounir Frikha
Department of Computer Networks and Communications-College of Computer Sciences and Information Technology,
King Faisal University, Al-Ahsa 31982, Saudi Arabia

***Abstract*—This paper examines the enhancement of security measures for the Internet of Things (IoT) systems through the application of Machine Learning (ML) techniques. As the number of IoT devices continues to rise, ensuring their security has become increasingly critical, given that conventional methods frequently struggle to identify advanced threats. This study explores the implementation of several ML algorithms, including Random Forest (RF), Decision Trees (DT), Support Vector Machines (SVM), and Convolutional Neural Networks (CNN), to identify anomalies and intrusions within IoT networks. By conducting a comprehensive review of existing research and experiments, it highlights the effectiveness of ML in enhancing IoT security, with high detection rates for various threats, including botnet attacks, Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS) incidents, and intrusion attempts. DoS/DDoS attacks and many types of botnets are the most devastating attacks that have been spreading for a long time, and they are still branching out in new ways against IoT networks. They can damage IoT services and prevent these services from being used by legitimate users. Therefore, securing IoT networks becomes a significant concern. The proposed model is used to increasingly monitor network traffic for any deviations from standard patterns IoT networks. This paper also stresses the necessity of utilising suitable datasets and feature selection techniques to enhance the efficacy of ML models. To train our model, we have utilized a dataset called the IoT23 dataset, which is one of the most recent datasets that has many IoT scenarios and anomalous activities. Furthermore, we utilised two types of feature selection algorithms, the Correlation-based Feature Selection (CFS) algorithm and the Genetic Algorithm (GA), and then we compared the results of these algorithms when training our model. The best performances were obtained with DT and RF classifiers when they were trained with features selected by CFS However, for training and testing times metrics, DT performance was superior across both feature selection methods.**

Keywords*—*Internet of Things; Artificial Intelligence; machine learning; deep learning; security

I. Introduction

The Internet of Things (IoT) is an innovative technology that integrates various solutions across a wide range of applications [1], [2]. IoT refers to a distributed network comprising multiple sensor devices and systems, such as QR code devices, RFID devices, and sensor networks [1]. However, with the rapid growth and easy accessibility of these smart devices and networks, the focus on IoT security has increased due to malicious users, leading to a surge in related challenges [2], [3]. Furthermore, traditional techniques are less effective at detecting various types of attacks [3], [4], which requires the development of new technological advancements [2]. Artificial Intelligence (AI) is a computing technology that improves the ability of the IoT to deliver advanced application services [5]. IoT devices can leverage AI technology, encompassing two types: Machine Learning (ML) and Deep Learning (DL), each with its own set of algorithms and classifiers [6], [5]. IoT employs a multilayer architecture, with one architecture relying on three layers: application, network, and perception layers [1].

A. Problem Statement

In today's world, the lack of IoT security in systems, devices, and networks against attacks and malicious actions by intruders is a significant concern. Despite this, it is challenging to secure IoT devices with traditional security controls, and existing solutions such as the cloud suffer from centralisation and high delay. Another contributing factor is the vulnerability of IoT devices. Furthermore, the lack of security standards has added another dimension to the complexity of securing IoT devices. These challenges call for a monitoring system, such as anomaly detection, at both the device and network levels beyond the organisational boundaries. In recent years, the use of ML techniques has increased, and it is used to develop anomaly-based approaches to protect IoT networks, systems, and applications. It has impressive results while trained on normal and abnormal data in order to detect anomalies. However, building effective and efficient anomaly detection modules is not easy at all. ML models require significant effort, and selecting the best-performing algorithms remains a major challenge. Selecting the best dataset for a system is another challenge because there are many datasets that are available in public, and it isn't easy to choose the best one that fits your needs. Some of the data sets are large and outdated, so in this research, we aimed to enhance IoT security using ML techniques that can detect and mitigate cyber threats over a suitable data set for the IoT environment.

B. Contribution

Our contributions in this paper are summarized as follows:

- Designing a robust anomaly detection framework for the IoT, which facilitates and improves the detection of malicious data.
- Proposing an efficient model to extract relevant IoT features and eliminate unnecessary ones, thereby improving anomaly detection performance.
- Evaluating the effectiveness of the proposed model using recent IoT datasets and comparing it with existing ML models used for anomaly detection.

C. Research Questions

- What research has been done on ML for IoT security?
- What methods have been tried and how effective were they?
- What are the limitations or challenges of previous studies?
- How can ML improve threat detection in IoT networks?
- How can feature selection improve ML-based IoT security models?
- What are the most suitable datasets to enhance IoT security using ML algorithms?
- What are the research gaps and future directions in IoT security?
- What is the impact of processing stages-include data preprocessing, feature selection, and sampling-on evaluating the classifier performance in detecting attack types?

This paper is organized as follows. Section II presents the background. Section III reviews previous research studies related to IoT security using ML technology or presents and analyzes the related work. Section IV presents the methodology. Section V discusses the producers of the model setup, including preparing the dataset and training the model with classification algorithms. Section VI presents and analyzes the experimental results. Section VII presents a comparison between our proposed approach and an existing study. Section VIII concludes the research and suggests future work.

II. Background

A. IoT Security Challenges

1) Common security challenges include:

- Weak authentication and access control → Many IoT devices have default passwords or weak credentials.
- Intrusion and malware attacks → IoT networks are vulnerable to cyber threats, such as Distributed Denial-of-Service (DDoS), botnets, and ransomware.
- Data breaches and privacy issues → Sensitive user data (e.g. health or financial data) can be intercepted.
- Lack of real-time threat detection → Traditional security methods (e.g. firewalls, rule-based detection) struggle to detect new and evolving threats in real time.
- Resource constraints in IoT devices → Many IoT devices have limited processing power, making complex security algorithms difficult to implement.

B. AI Technology for IoT Security

Artificial Intelligence (AI), including ML and DL, is an effective solution to increase system security with regard to IoT. Traditional solutions require more processing ability for large datasets and are less efficient than AI technology. The remarkable ability of learning models to solve long-standing problems has made them increasingly important in recent times. Another significant advantage of learning algorithms is their ability to improve accuracy through real-time learning from test data. This section reviews various ML algorithms, how they function, and their contributions to IoT network security.

C. Machine Learning in IoT Security

Recent research shows that ML models significantly improve intrusion detection, anomaly detection, and malware classification. Studies have explored Convolutional Neural Networks (CNNs) for traffic classification, Long Short-Term Memories (LSTMs) for sequential anomaly detection, and Federated Learning for decentralised security. However, a trade-off between accuracy and computational efficiency remains a challenge.

1) Role of machine learning in IoT security: ML has emerged as a promising solution for securing IoT environments. ML-based security models can detect, predict and respond to cyber threats by learning patterns in network traffic. Unlike traditional security approaches, ML can handle large datasets, adapt to evolving threats, and minimize false positives. Various ML techniques have been applied in IoT security, including supervised learning, unsupervised learning, and deep learning models.

2) Key advantages of ML in IoT security:

- Anomaly Detection: ML can detect suspicious patterns in IoT traffic and identify potential cyberattacks.
- Adaptive Security: Unlike rule-based systems, ML continuously learns and updates itself to detect new threats.
- Low False Positive Rate: ML reduces false alarms compared to traditional signature-based detection.
- Scalability: ML can efficiently analyze large networks.
- Automated Threat Detection: ML models can work in real time, preventing attacks before they escalate.

Table I below briefly describes the K-means clustering and Q-Learning methods that belong to the RL model.

III. Related Work and Literature Analysis

This section reviews previous research on IoT security using ML. The studies analysed focus on various ML models, datasets, and techniques to enhance IoT security.

Nazir et al. [6] proposed a novel Collaborative Threat Intelligence Framework that integrates blockchain and ML for enhanced IoT security. The study used Random Forest (RF), Decision Trees (DT), Ensemble Learning, LSTM, and CNN models, using the IoT23 dataset. The primary issue addressed is the dynamic and evolving nature of IoT threats, which traditional security methods often fail to mitigate. The key contribution is a real-time threat detection system that improves accuracy by reducing false negatives through collaborative intelligence. Although the results showed improved

TABLE I. LEARNING MODELS

Learning model	Classifiers	Principle	Advantages	Disadvantages	Applications applicable to IoT
Supervised learning	SVM	Generates classified data by establishing a hyperplane between two or more classes. Expands the margin around the hyperplane.	• Stable in handling datasets with numerous features	• Inefficient with very large datasets • Struggles with multi-classification scenarios	• Intrusion detection • DoS/DDoS detection • Malware detection
Supervised learning	DT	Splits datasets into branches based on feature values, forming a tree structure. Branches represent decision rules ending in outcomes.	• Quick computations • Allows intuitive decision making	• Becomes overly complex with multi-dimensional data	• Intrusion detection • DoS/DDoS detection • Device authentication
Supervised learning	KNN	Assigns classifications based on the nearest sample votes.	• Straightforward and adaptable • Effective for both non-linear and multi-class scenarios	• Time-consuming to optimize K value	• Intrusion detection • DoS/DDoS detection
Supervised learning	ANN	Mimics a biological neural network using sample weights and biases.	• Handles incomplete data effectively • Capable of managing complex models due to hidden layers	• High computational demand • Challenges in tracing predictions	• Intrusion detection • DoS/DDoS detection • Malware analysis • Device authentication
Supervised learning	NB	Assumes independence among input variables	• Usually applicable for both small-scale data and large-scale data • Simple training process	• Independence assumption often not met	• Intrusion detection • Device authentication
Supervised learning	RF	Utilizes multiple decision trees, each trained on random data subsets.	• Better than single-model • Difficult to overfit • Excellent noise resilience • Works with high-dimensional data • Handles both continuous and discrete data	• Potential for overfitting • Degrades with noisy data	• Intrusion detection • DoS/DDoS detection
Unsupervised learning	PCA	Transforms correlated features into fewer uncorrelated ones.	• Simplifies models • No parameter constraints	• Less interpretable components • Sensitive to scaling and rotation	• Feature selection • Feature detection
Unsupervised learning	K-means Clustering	Groups samples into K clusters based on feature similarity.	• Efficient anomaly detection • Useful for data anonymisation	• May lack accuracy • Not ideal for complex data structures	• Network normalization • Sybil detection in WSNs • Private data anonymisation
Reinforcement learning	Q-Learning	Optimizes decisions in IoT through trial and error learning.	• Requires few parameters • Can be implemented offline	• High memory and time cost • Not ideal for high-dimensional data	• Intrusion detection • DoS/DDoS detection • Malware detection • Device authentication

accuracy and adaptability in IoT security applications, reliance on blockchain increased computational overhead.

Kirana et al. [7] developed an Intrusion Detection System (IDS) using Naïve Bayes (NB), SVM, DT, and AdaBoost classifiers. The testbed simulated IoT environments with ESP8266 and DHT11 sensors, collecting network traffic for ML model training. The study focused on detecting Man-in-the-Middle (MITM) attacks and demonstrated high accuracy in anomaly classification. However, adversarial attacks on ML models remain an unresolved challenge.

Alwahedi et al. [8] provided a comprehensive survey on ML-based Intrusion Detection Systems (IDS), including DL models such as Artificial Neural Networks (ANN), SVM, RF, and DT. The key issue tackled is the heterogeneity of IoT security threats. A significant gap identified is the lack of real-time response mechanisms in existing IDS solutions. The proposed solution integrates Generative AI and large language models (LLMs) for adaptive cyber threat mitigation.

Zawaideh and Abu Baker [9] introduced a cascaded ML approach for detecting botnet attacks in IoT networks. The study utilised a Recurrent Neural Network (RNN) combined with LightGBM, trained on the BOT-IoT dataset. The proposed model outperformed traditional classifiers with an accuracy of 99.9% and a recall of 100%, effectively handling imbalanced IoT traffic. The main limitation was the higher computational complexity of the cascaded model.

Bagaa et al. [10] developed an ML-driven security framework leveraging Software-Defined Networking (SDN) and Network Function Virtualisation (NFV). The framework employs One-Class SVM, Distributed Data Mining, and Neural Networks for real-time anomaly detection in innovative building IoT environments. Experimental results showed a 99.71% detection accuracy is achieved, although scalability issues in edge computing environments remain a challenge.

Al-Garadi et al. [3] presented a comprehensive review of ML and DL techniques for IoT security. The research

emphasised the effectiveness of DL models like CNNs and RNNs for detecting cyber threats. A key limitation highlighted is the lack of labelled datasets for IoT-specific threats, making supervised learning less effective in real-world scenarios.

Boukerche and Coutinho [11] outlined best practices for implementing ML-based security mechanisms in IoT networks. The study focused on optimising computational efficiency while maintaining high detection accuracy. The trade-off between detection accuracy and computational cost remains a primary challenge in real-world deployments.

Makkar et al. [12] introduced an ML-based spam detection framework for IoT devices. The study evaluated five ML models, using the REFIT Smart Home dataset. The best-performing models demonstrated high accuracy in detecting anomalous behavior, though the framework struggles with adversarial ML attacks.

Dhanke et al. [13] developed an ML-based IDS using RF, DT and SVM models. The study focused on Cyber-Physical Systems (CPS) and Smart Home environments, using ML to detect abnormal network traffic and malicious activities. The key findings revealed that ML improved security by detecting intrusions efficiently, achieving 85% accuracy. However, the study identified a gap in real-time adaptability, as some ML models struggled with evolving threats.

Gad et al. [14] introduced a distributed IDS leveraging RF, XGBoost, and LightGBM for anomaly detection in IoT networks. The study utilised the ToN-IoT dataset, which includes diverse cyberattacks. The key finding was that distributed ML models enhanced intrusion detection. However, high resource consumption on edge devices remained a limitation. The proposed lightweight IDS aimed to balance security and computational efficiency.

Ahmad and Almada [4] conducted a systematic literature review on the role of ML and DL techniques in IoT security. The review identified that supervised learning methods were more effective for known attack detection, while unsupervised learning performed better in zero-day attack scenarios. A key challenge highlighted was the scarcity of high-quality, labelled datasets, which limits the effectiveness of ML-based security models.

Amanullah et al. [5] investigated the use of DL for IoT security by analysing cyber threats using CNN and RNN. The research proposed a big data-driven approach for handling large-scale security threats. The study concluded that DL models significantly improved detection accuracy; however, the high computational overhead limited their deployment in low-power IoT devices.

Alrowais et al. [15] introduced an Automated Machine Learning (AutoML) approach using Mayfly Optimisation (MFO) with Regularised Extreme Learning Machine (RELM) for IoT security threat detection. The research used real-world IoT security datasets, demonstrating superior performance in detecting various attack types. The main limitation was the difficulty in real-time adaptation of AutoML models.

Awajan. [16] Introduced a DL-based IDS using a fully connected neural network. The study focused on detecting Blackhole, DDoS, and Sinkhole attacks with an average accuracy of 93.74%. A major challenge identified was the lack of communication protocol-independent security models, which limited interoperability across different IoT systems.

Tahir et al. [17] introduced an ML-based anomaly detection system to strengthen IoT security. The study employed RF, DT, SVM, and Gradient Boosting models to analyse IoT network traffic data. The key problem addressed was adaptive security mechanisms for anomaly detection in IoT environments. The research found that Gradient Boosting achieved the highest precision (89.34%), demonstrating the effectiveness of ML in securing IoT systems. However, a significant limitation identified was computational inefficiency, making real-time threat detection difficult in resource-constrained IoT environments.

Musleh et al. [18] proposed an IDS using feature extraction with ML algorithms. The study tested RF, KNN, SVM, and stacking models combined with feature extraction techniques such as VGG-16 and DenseNet. The dataset used was the IEEE Data Port, and the key finding was that VGG-16 with stacking achieved 98.3% accuracy, demonstrating the potential of DL for intrusion detection in the IoT. However, the reliance on large labelled datasets and computational overhead remains a significant challenge.

Khatun et al. [19] reviewed the application of ML in securing Healthcare Internet of Things (H-IoT). The study focused on intrusion detection, authentication, and anomaly detection using RF, DT, and Gradient Boosting models. The key finding was that ML algorithms significantly improve healthcare security, but privacy concerns and adversarial attacks on IoT devices remain a challenge. The study proposed a robust authentication mechanism using ML but highlighted that real-time data analysis remains computationally expensive.

Ahmad et al. [20] conducted a comprehensive review on ML and DL techniques for IoT security. The study highlighted the effectiveness of supervised models for known attack detection, while unsupervised learning was better suited for zero-day threats. A key challenge was the lack of labelled datasets, limiting the effectiveness of ML-based security models.

Mukherjee. I et al. [21] proposed a model of two different approaches to predict anomalies in IoT networks using a 350k-record dataset and deploying ML models. The study used Logistic Regression (LR), DT, RF, NB, and ANN algorithms first with all features, then with selected features after preprocessing. The models achieved 99.4% accuracy with all features and 99.99% after feature selection. It categorises attacks into DoS, malicious control, malicious operation, incorrect setup, and spying. The model was trained and tested on all attack types, rather than only common ones.

Bedi. P et al. [22] presented four models based on ML technology; these methods are ANN, LR, RF, SVM and DT. The paper aims to assess the results of all these models via evaluation metrics to find the best performance of the ML algorithm in detecting attacks in IoT sensor networks. The proposed IoT framework, structured by dataset perception, assortment and information processing, was executed on the dataset as an essential stage. Furthermore, the dataset used is known as the DS2OS dataset, which contains eight classes and seven attack types, including DoS attacks, data probing, malicious control, malicious operation and spying. The results showed that ANN and DT classifiers have better performance than LR, EF and SVM.

Altulaihan.E et al. [23] proposed a new mechanism using IDS to enhance the security of IoT networks. The paper focused on mitigating a type of attack called a DoS (Denial of Service) attack through four supervised classifier algorithms: KNN, SVM, DT, and RF, within an existing dataset known as the IoTD20 dataset, achieving an accuracy of 99.9%. Other datasets mentioned were CIDDS-001, UNSW-NB15, NSL-KDD, Bot-IoT, and KDD99. The results showed that DT and RF were the most efficient methods compared to others.

Alotabib et al. [24] Proposed a stacked DL approach for IoT security. This method aims to detect malicious traffic data, including phishing, DDoS, spamming campaigns, and data leakage. The proposed approach utilises pre-trained residual networks (ResNets), which help discover the characteristics of activities, whether normal or abnormal. The experiment has been done by using two cases using two datasets, including the N-BaIoT dataset and the power system dataset. Furthermore, the two scenarios target two heterogeneous IoT environments: smart homes and smart grids. NB, SVM and DT are ML algorithms that have been used for the proposed model.

Trcˇek.D et al. [25] presented a new anomaly detection system that is based on appropriate lightweight ML algorithms known as Profile and Hierarchical Incremental Clustering-based Anomaly Detection (PHICAD). The PHICAD enhances security, which works on the assumption of a large number of frequent flows that include normal activities and abnormal activities, and a small number of infrequent ones. Furthermore, it enables detection of unknown and new anomalies. Furthermore, the flows are arranged into profiles and each represents a model of network activity it assigned into a single network entity and it has it won IP address. This paper has been evaluate PHICAD by using CIC-IDS-2017 which contains most of network cybersecurity attacks (web attacks, brute force, port scan, heartbleed, DDoS, DoS, botnet and infiltration). Other ML algorithms performance have been compared such as KNN, RF, ID3 and NB using CIC-IDS-2017 dataset and result showed that the proposed approach achieved the highest precision rate.

A. Key Findings and Insights

Tables II and III presents a compilation of research studies that explore the application of ML techniques to enhance security in IoT environments.

In Tables II and III, we have highlighted several significant findings that enhance IoT security through the application of ML models. A key observation is the variety of ML models employed in various studies to tackle a broad spectrum of IoT security threats. These models range from traditional approaches such as DT, RF, SVM, KNN, and NB, to more sophisticated DL techniques including ANN, CNN, and RNN.

Additionally, some research incorporated hybrid or automated solutions like LightGBM, AutoML, and Generative AI. The effectiveness of each model varied depending on the specific security challenge—such as intrusion detection, DoS/DDoS attacks, botnet identification, and anomaly detection—indicating that a model's success is closely tied to the type of attack and the characteristics of the dataset. Furthermore, a notable insight is the consistent high detection accuracy reported across numerous studies. For example, Zawaideh and Abu Baker achieved 99.9% accuracy and 100% recall in identifying botnet attacks using a hybrid RNN and LightGBM method, while Bagaa et al. reported 99.71% accuracy in a smart building scenario using a One-Class SVM model. These results underscore the capability of well-trained ML models to provide reliable and accurate detection of cyber threats within IoT settings. However, despite these encouraging outcomes, challenges persist, particularly in realizing real-time detection and adaptability in dynamic, resource-limited IoT networks. As highlighted by Gad et al., the high computational requirements, especially at the edge level, can impede the deployment of distributed ML-based anomaly detection systems. This underscores the necessity for lightweight, adaptive models that can sustain performance in the face of evolving threats. The quality and selection of datasets are vital for the performance of models. Popular datasets like IoT23, BOT-IoT, and CIC-IDS-2017 offer extensive attack scenarios, but relying on outdated, unbalanced, or simulated data can limit the generalizability of studies. This highlights the need for choosing or creating datasets that accurately represent real-world IoT security situations. Feature selection also plays a crucial role in the success of models.

Many studies point out the advantages of employing feature selection techniques, such as PCA or correlation-based methods, to minimize dimensionality and enhance both accuracy and efficiency. Effective feature engineering is essential for optimizing model architecture and preventing overfitting. Furthermore, a significant issue that has arisen is the susceptibility of ML models to adversarial attacks. Malicious actors can alter inputs to trick models into making incorrect classifications. This necessitates the creation of robust models that are aware of adversarial threats and the incorporation of privacy-preserving strategies, particularly due to the sensitive information managed by IoT devices.

Finally, the balance between model complexity and computational efficiency continues to pose a challenge. Although DL models typically provide greater accuracy, their substantial resource requirements can hinder their use in real-world, low-power IoT environments. Hybrid strategies that merge lightweight traditional ML techniques with DL elements present a promising way to achieve a balance between detection quality and practical application.

Fig. 1 illustrates the accuracy percentages of different ML models across various studies as presented in Table II. Each segment of the pie represents the accuracy achieved in different studies, highlighting the performance of the models used in IoT networks.

After analyzing Table II, we suggest using ML models such as RF, DT, and CNN for IoT security. These models have shown strong performance in detecting a wide range of IoT security threats, including intrusion detection, DoS/DDoS attacks, botnet detection, and anomaly detection. RF and DT are particularly useful for their robustness, ease of implementation, and good performance on structured data. CNNs, on the other hand, are beneficial for more complex attack detection scenarios, particularly those involving network traffic and sequential data patterns.

In terms of datasets, we recommend using IoT23. This dataset provides a wide range of IoT-specific attack scenarios

TABLE II. SUMMARY OF EXISTING STUDIES ON ML TECHNIQUES IN IOT SECURITY

Reference	ML Models Used	Problem Addressed	Datasets Used	Results
Nazir et al. (2024) [6]	RF, DT, Ensemble Learning, LSTM, CNN	Evolving IoT threats and ineffective traditional security	IoT23 dataset	Enhanced real-time threat detection, reduced false negatives
Kirana et al. (2020) [7]	Naïve Bayes, SVM, DT, AdaBoost	MITM attack detection in IoT environments	Simulated IoT environment (ESP8266, DHT11 sensors)	High accuracy in detecting MITM attacks
Alwahedi et al. (2024) [8]	ANN, SVM, RF, DT	Heterogeneous IoT security threats and lack of real-time response	Not specified	Proposed Generative AI and LLMs for adaptive threat mitigation
Zawaideh and Abu Baker (2023) [9]	RNN, LightGBM	Botnet attack detection in IoT networks	BOT-IOT dataset	99.9% accuracy, 100% recall
Bagaa et al. (2020) [10]	One-Class SVM, Distributed Data Mining, Neural Networks	Real-time anomaly detection in smart building IoT	SDN/NFV-based IoT environments	99.71% detection accuracy
Al-Garadi et al. (2020) [3]	CNN, RNN	Cyber threat detection using ML/DL	Various IoT security datasets (review study)	DL models effective for IoT security
Boukerche & Coutinho (2020) [11]	ML-based security mechanisms	Optimizing computational efficiency while maintaining detection accuracy	Not specified	Trade-off between detection accuracy and computational cost remains a challenge
Makkar et al. (2020) [12]	Evaluated five ML models	Spam detection for IoT devices	REFIT Smart Home dataset	High accuracy in detecting anomalies, but vulnerable to adversarial ML attacks
Dhanke et al. (2021) [13]	RF, DT, SVM	Intrusion detection in CPS and Smart Homes	Not specified	85% accuracy, but struggled with evolving threats in real-time adaptability
Gad et al. (2022) [14]	RF, XGBoost, LightGBM	Distributed IDS for IoT anomaly detection	ToN-IoT dataset	Enhanced intrusion detection, but high resource consumption on edge devices
Ahmad & Alsmadi (2021) [4]	Supervised and Unsupervised Learning	ML and DL techniques in IoT security	Various datasets	Supervised learning best for known attacks, unsupervised better for zero-day attacks; lack of high-quality labeled datasets remains a challenge
Amanullah et al. (2020) [5]	CNN, RNN	Analyzing cyber threats in IoT using DL	Not specified	Improved detection accuracy, but high computational overhead limits deployment in low-power IoT devices
Alrowais et al. (2022) [15]	AutoML (MFO-RELM)	Automated ML for IoT security threat detection	Real-world IoT security datasets	High accuracy in detecting various attack types; difficulty in real-time adaptation
Awajan (2023) [16]	Fully connected neural network (DL-IDS)	Intrusion detection (Blackhole, DDoS, Sinkhole attacks)	Not specified	93.74% accuracy; challenge in developing protocol-independent security models
Tahir et al. (2024) [17]	RF, DT, SVM, Gradient Boosting	Adaptive anomaly detection in IoT security	IoT network traffic data	Gradient Boosting achieved 89.34% precision; computational inefficiency limits real-time detection
Musleh et al. (2023) [18]	RF, KNN, SVM, Stacking (VGG-16, DenseNet)	Feature extraction for IDS	IEEE Dataport	VGG-16 with stacking achieved 98.3% accuracy; challenges in dataset labeling and computational overhead
Khatun et al. (2023) [19]	RF, DT, Gradient Boosting	ML for Healthcare-IoT (H-IoT) security	Not specified	ML improves healthcare security; challenges in adversarial attacks and real-time computational efficiency
Ahmad et al. (2021) [20]	Supervised and Unsupervised Learning	ML and DL techniques for IoT security	Various datasets	Supervised models effective for known attacks, unsupervised better for zero-day threats; challenges in dataset availability
Mukherjee.I et al. (2020) [21]	LR, DT, RF, ANN, NB	Detect IoT threats	DS2OS traffic traces	Most classifiers achieved 99.9% accuracy except RF with 94%, indicating lower performance

and the latest real-world data, making it ideal for training and testing ML models in the context of IoT security. IoT23 is well-suited to detect a variety of network attacks and provides comprehensive coverage for anomaly detection and botnet identification. On the other hand, many mentioned papers ether used outdated datasets, lacked most recent attack types, complex dataset, lacked of IoT traces, and lacked contained features related to IoT.

These ML models and datasets, when combined, will enable more effective and robust detection of IoT security threats, enhancing the overall security posture of IoT networks.

IV. METHODOLOGY

ML classification algorithms were used to detect anomalies in IoT networks. This approach monitors network traffic and data flows for deviation from normal network profiles based on anomaly detection. This methodology consists of several key stages, as shown in Fig. 2. First is the processing phase, where the preferred dataset is selected to train ML classification algorithms. Preprocessing is then performed to clean and transform the data. Null and irrelevant values are removed, as they are difficult to handle and may lead to incorrect results. Second is the feature selection phase, which includes preprocessing the features. Third, the dataset is divided into two subsets: the training subset and the test subset. By selecting the right testing and training data, classification accuracy can be improved. The training data are used to train the model, while the test data evaluate the model's performance. Finally, classification identifies whether the input belongs to a normal class or a specific type of attack using selected ML algorithms. We will

TABLE III. SUMMARY OF EXISTING STUDIES ON ML TECHNIQUES IN IOT SECURITY

Reference	ML Models Used	Problem Addressed	Datasets Used	Results
Khatun et al. (2023) [19]	RF, DT, Gradient Boosting	ML for Healthcare-IoT (H-IoT) security	Not specified	ML improves healthcare security; challenges in adversarial attacks and real-time computational efficiency
Ahmad et al. (2021) [20]	Supervised and Unsupervised Learning	ML and DL techniques for IoT security	Various datasets	Supervised models effective for known attacks, unsupervised better for zero-day threats; challenges in dataset availability
Mukherjee I. et al. (2020) [21]	LR, DT, RF, ANN, NB	Detect IoT threats	Ds2OS traffic traces	All classifiers except RF achieved 99.9% accuracy; RF achieved 94%, indicating lower performance
Bedi P. et al. (2021) [22]	ANN, RF, LR, DT, SVM	Assessment of ML algorithms to select best for attack detection in IoT sensor networks	DS2OS dataset	DT and ANN classifiers achieved high precision rates around 99%
Altulaihan E. et al. (2024) [23]	DT, RF, KNN, SVM	Detection of DoS attacks in IoT networks	IoTID20 dataset	DT and RF achieved 100% detection performance
Alotabib B. et al. (2024) [24]	NB, DT, SVM	Cyberattack detection against IoT devices	Power system dataset and N-BaIoT dataset	RF achieved highest accuracy: 100% and 99.9%, respectively
Trček D. et al. (2024) [25]	PHICAD	New lightweight ML approach to detect various anomalies in IoT network	CIC-IDS-2017 dataset	Achieved 99% precision, 84% recall, and 91% F1-score

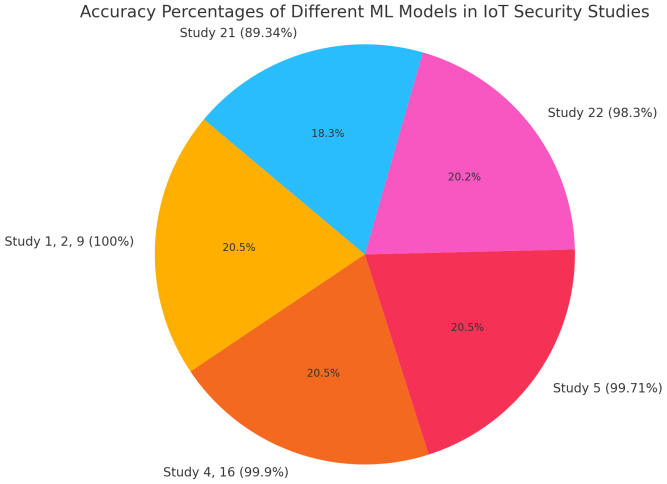


Fig. 1. The accuracy percentages of different ML models.

validate our proposed scheme using a confusion matrix.

A. Evaluation and Analysis

The evaluation and analysis process includes the following steps:

- Evaluate the performance of the IDS in simulated or real-world environments.
- Analyze the system's performance using metrics such as detection rate, false positive rate, and response time.

B. Optimization and Improvement

The optimization and improvement phase includes the following steps:

- Identify potential areas for improvement in the attack detection system based on the evaluation results.
- Conduct further research and development to optimize the system's performance and enhance its robustness against evolving attack techniques.

C. Identification and Analysis of Alternative Solutions

Multiple alternative solutions exist for implementing the system, and various methods can be used. In this section, we explain each solution.

1) Feature selection algorithms: Two different feature selection algorithms were used to improve detection accuracy and the speed of identifying malicious activities. The process involves both selecting the most relevant features and removing redundant and irrelevant features. Additionally, we compared the performance of Genetic Algorithm (GA) and Correlation-Based Feature Selection (CFS) methods.

- GA is a natural selection-based optimization technique that involves several steps. It begins with creating an initial population of possible features, followed by evaluating these subsets through a predictive model. Next, subsets are selected to continue based on tournament selection, where each member competes for survival. In the next generation, winners undergo crossover and mutation to form new subsets of features.
- CFS is a filter-based approach that evaluates the correlation between input and output features. It eliminates redundant or irrelevant features that could increase computation time or reduce detection accuracy.

Table IV provides a comparison of the advantages and disadvantages of both GA and CFS algorithms.

TABLE IV. ADVANTAGES AND DISADVANTAGES OF GA AND CFS ALGORITHMS

Algorithm	Advantages	Disadvantages
GA	• Finds good solutions quickly and accurately • Easy to implement	• Difficult parameter selection • May not yield optimal solutions
CFS	• Lower computational complexity than GA • Strong predictive power for gene selection • Less prone to overfitting	• May not fit data well due to model dependency

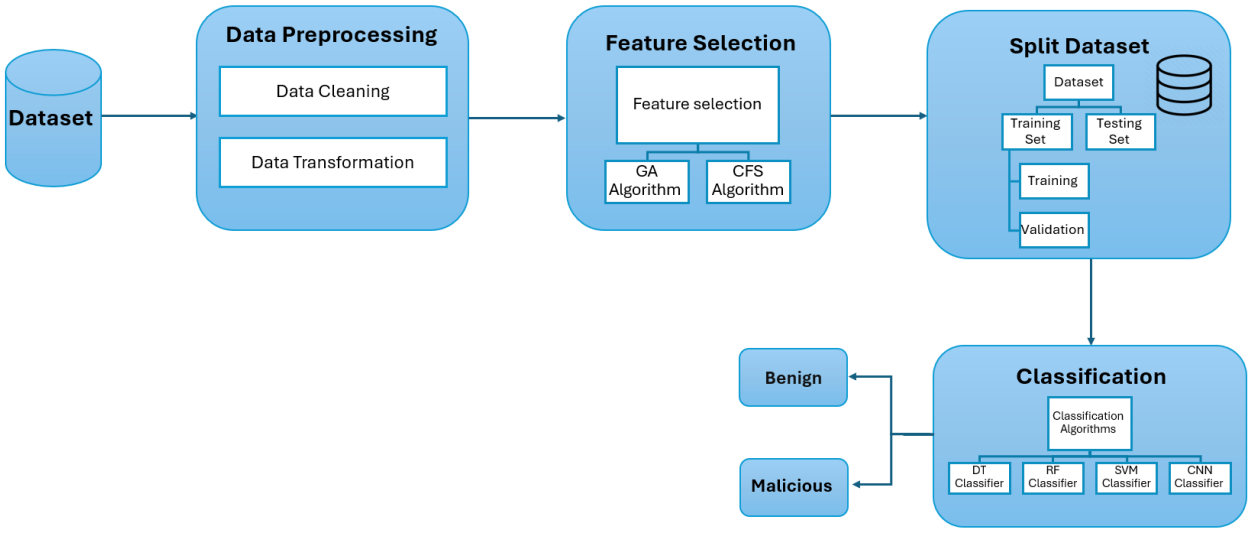


Fig. 2. Evaluation of the performances.

2) Machine learning algorithms solutions and selected dataset: We applied four machine learning algorithms to the IoT23 dataset: DT, RF, SVM, and CNN. In Table I, we clarify the advantages and disadvantages of the four classifiers.

3) Performance metrics: After implementing and validating the model, we conducted testing to assess its performance. The initial evaluation focuses on accuracy and timeliness in predicting data. Subsequently, the model's training and testing performance are evaluated using metrics such as accuracy, precision, and recall Performance metrics are systematic measures that help assess how well a tool performs. In ML, these metrics are used to evaluate the effectiveness of features, feature selection methods, and ML algorithms. Performance metrics play a crucial role in assessing the effectiveness of four types of ML algorithms: DT, RF, SVM, and CNN. These metrics help evaluate the classification performance of the feature set and different ML algorithms. Some fundamental metrics include True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These basic metrics serve as building blocks for calculating more advanced metrics such as precision, recall, accuracy, F-measure, and AUC. The key advanced metrics considered are precision, accuracy, recall, and F1-score.

- Accuracy: The accuracy formula is as follows.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- Precision: The precision formula is as follows.

$$\text{P}recision = \frac{TP}{TP + FP}$$

- Recall: The recall formula is as follows.

$$\text{R}ecall = \frac{TP}{TP + FN}$$

- F-measure: The F-measure formula is as follows.

$$\text{F} - measure = 2x\frac{precision + recall}{precisionxrecall}$$

V. Setup of the Model

To implement the IDS, the model is structured in two main stages: dataset preparation and training with classification algorithms.

A. Data Collection

The IoT23 dataset was selected for this study. It contains 23 PCAP files used in various use cases such as intrusion detection, malware analysis, traffic classification, and anomaly detection in IoT networks. The dataset contains 28 columns called features including duration, orig_bytes, resp_bytes, missed_bytes, orig_pkts, orig_ip_bytes, resp_pkts, resp_ip_bytes, and label. The model focuses on the 'label' feature, which indicates whether network traffic is benign or malicious.

1) Load dataset: To train the system for detecting malicious activity in the IoT network, we used the IoT23 dataset due to its comprehensive feature set. First, we downloaded the dataset from its official website and saved it locally. Then, we loaded the dataset by locating the path to our dataset file, storing it in the variable "path," and assigning the dataset to the data frame object "df" with the help of the Pandas library. Finally, we printed the dataset using the df.head() function.

B. Dataset Preprocessing

In this stage, we enhance the dataset and make it more specific by performing a series of manipulations, including cleaning, encoding, scaling, and noise removal.

1) Cleaning: In the data cleaning process, we removed null values and their corresponding entries. Missing values can be challenging to handle for ML algorithms, potentially leading to incorrect predictions[26]. We utilized the isnull().sum() function to check for null values.

2) Encoding: First, we used the "df['label'].value_counts()" function to determine how frequently benign and malicious packet traffic appeared. Second, as included in the dataset, it classified them as benign (normal) or attack types (Attack, PartOfAHorizontalPortScan, PartOfAHorizontalPortScan, C&C, C&C-FileDownload, C&C-HeartBeat, C&C-HeartBeat-FileDownload, C&C-Mirai, C&C-Torii, DDoS, FileDownload, Okiru, and PartOfAHorizontalPortScan). From these attack types, we selected only those relevant to our study—specifically intrusion, DDoS, and botnet attacks. Finally, we converted all labels to 1 or 0 using the "lambda" function and the "apply" method.

Our dataset is noise-free, as it was generated through the simulation of our system. Therefore, there is no need for noise removal steps.

C. Feature Selection

We must identify which features are important for the training and testing phases and eliminate those that are unnecessary. GA is an algorithm used for feature selection. The GA is an optimization algorithm using the wrapper method which starts with a set of points coded as an alphabet rather than one real parameter set. Additionally, there is no need for a calculation step size or derivative information in GA. When applied to our dataset, GA selected 14 out of 28 features as the most relevant for training the system.

CFS was also employed as an alternative feature selection method. Features are selected and analyzed based on their relationship with the target variable and their interrelationships. After executing the CFS algorithm, the top-ranked features were prioritized for application within our dataset.

D. Split the Dataset into Training and Testing Sets

The dataset was divided into two subsets: training and testing. This is a crucial step in the ML process, as it directly impacts the model's accuracy. The training data were used to fit the model, while the testing data evaluated its performance. Then, the testing data is used to evaluate the performance of the model. We used the train_test_split() function to divide the dataset into training and testing sets with a 67/33 split, using random_state=42 for reproducibility. We set the test size to 33% of the dataset and the remaining 67% was used for training. In addition, we used the "random_state" parameter with a value of 42 to ensure reproducibility. After splitting the dataset, we applied the Synthetic Minority Over-sampling Technique (SMOTE) to baa lance the training data.

E. Training with Classification Algorithms

The most critical stage of our project involves training and testing the model using ML algorithms. Once the training and testing sizes were known, we worked with classification algorithms. We selected four classification algorithms—DT, RF, SVM, and CNN—based on their effectiveness of algorithms in distinguishing between benign and malicious traffic. This section compares and evaluates the four algorithms using performance metrics such as confusion matrix, accuracy, precision, recall, and F1-score. We then identified the best-performing algorithm among the four.

1) Decision tree: DT is a classification algorithm used to determine whether the traffic is an attack or not. It has a tree-like structure that is easy to understand and mimics human decision-making. It consists of internal decision nodes and leaf nodes for outcomes, working by recursively splitting the dataset using attribute selection measures. In this study, the DT model was trained using the GA algorithm with Scikit-Learn's DecisionTreeClassifier.

2) Random forest classifier training: The RF model was implemented by creating randomly selected subsets of the training data to generate multiple decision trees. The method is based on building multiple DTs from a randomly selected subset of the training set and then combining all votes for every tree to make a final prediction. Additionally, the RF model effectively identifies the class of each input instance. Each decision tree processes input data to determine the corresponding category. The RF generates multiple trees and then averages the predictions, which helps mitigate overfitting [26].

To implement the classifier, a random sample of size n is selected for building each decision tree. Two essential criteria are carried out: 1) m features are randomly selected from all features, and 2) the data is split based on the m features. A total of k trees are constructed, and the final prediction is obtained by averaging their outputs [26].

3) Support vector machine training: The SVM algorithm is primarily used for classification but can also handle regression problems and both continuous and categorical variables. SVM aims to separate classes by constructing a hyperplane in multidimensional space. Iterative SVM algorithms help minimize errors. The algorithm seeks to define a maximum marginal hyperplane to split the dataset into multiple classes. To implement SVM, we separate the dataset by identifying the margin—the distance between classes. This is known as the margin. The goal is to determine a hyperplane that offers the widest margin between support vectors [26].

We implemented the SVM model by importing it and then creating a support vector classifier object using an SVC(). The SVC() object includes the gamma parameter to control the influence of individual training examples. The model was trained using the train set with the fit() function and made predictions on the test set using the "predict()" function. After training, predictions were made on the test data, and the SVM model was compared with other classifiers.

4) Convolutional Neural Network (CNN): is a powerful DL algorithm, effective for both image classification and sequential or network traffic data like intrusion detection. CNN includes convolutional layers like Conv1D, which automatically extract important patterns from input data, followed by pooling and flattening layers. The dense layer, a key component, is responsible for performing the final classification. CNN was applied to enhance the detection of attacks within network traffic data.

In addition, by using learning deep feature representations we can improve classification accuracy.

5) Evaluate the performance: The evaluation phase was based on the confusion matrix. We compared the results of classifier algorithms with GA and CFS feature selection algorithms. Furthermore, to evaluate the classifier algorithms, we chose to use accuracy, precision, recall, and the F1 score.

VI. Results

This section presents and analyzes the experimental results, including the training and testing times of various classifiers, and evaluates their performance using key metrics such as accuracy, precision, recall, and F1-score.

A. Training and Testing Time

To evaluate the computational efficiency of the selected classifiers, we measured both training and testing times using two different feature selection techniques: GA and CFS. Table V and Fig. 3 present a comparative analysis of these times across the four classifiers: DT, RF, SVM, and CNN.

From the results, DT exhibited the fastest performance in both training and testing phases across both feature selection methods. Specifically, training with GA took only 0.6821 seconds, and testing took 0.0168 seconds. With CFS, the training and testing times increased slightly to 1.1308 and 0.0285 seconds, respectively.

RF demonstrated moderate computational costs, with GA yielding faster results compared to CFS. Notably, the training time for RF with GA was 33.4947 seconds, while testing took 1.1156 seconds. CFS led to slightly increased times at 43.9254 seconds (training) and 1.0211 seconds (testing).

SVM showed the highest variation in computational time. With GA, SVM required significantly higher resources, recording 5448.912 seconds for training and 1005.804 seconds for testing. However, with CFS, there was a remarkable reduction to 158.8664 seconds for training and 94.2506 seconds for testing.

On the other hand, CNN had a moderate-to-high training and testing time, with GA being faster than CFS. CNN took 188.9341 seconds for training and 6.1233 seconds for testing when GA was applied. Under CFS, training time increased to 387.9729 seconds, while testing time rose to 24.252 seconds.

In Fig. 3, GA generally yielded lower testing times, especially for DT and RF, making it suitable for time-sensitive environments. However, CFS demonstrated better training time efficiency for high-complexity models like SVM, highlighting its utility in scenarios where training overhead needs to be minimized.

From these results, we can conclude that the best-performing classifier in terms of both training and testing time is the DT algorithm. It consistently achieved the lowest execution times across both GA and CFS feature selection methods, making it an ideal choice for real-time IoT anomaly detection scenarios where computational efficiency is critical. This finding highlights the practicality of DT for deployment in lightweight and resource-constrained IoT environments.

TABLE V. Training and Testing Times (seconds) for Different Classifiers

Classifier	Training Time (s)		Testing Time (s)	
	GA	CFS	GA	CFS
DT	0.6821	1.1308	0.0168	0.0285
RF	33.4947	43.9254	1.1156	1.0211
SVM	5448.912	158.8664	1005.804	94.2506
CNN	188.9341	387.9729	6.1233	24.252

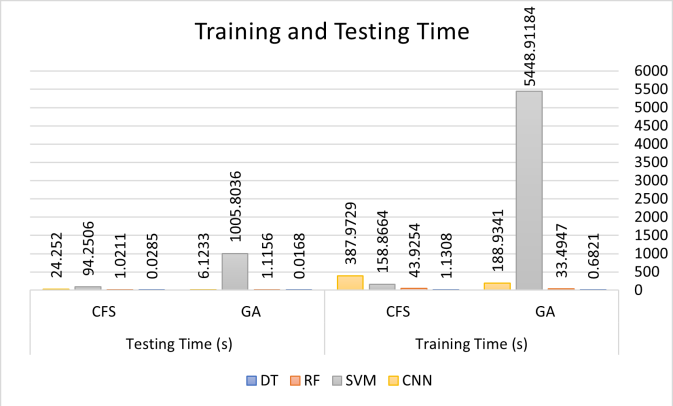


Fig. 3. Training and testing times.

1) Evaluate the performance: This section evaluates the performance of the classifiers using key metrics: Accuracy, Precision, Recall, and F1-score. The results are expressed in percentages for clarity, and both GA and CFS were applied for feature selection.

a) Accuracy: Accuracy evaluates the overall precision of the classifier. As illustrated in Table VI and Fig. 4, both the DT and RF attained the highest accuracy of 99.9% when utilizing features selected by CFS. SVM and CNN also exhibited remarkable performance, increasing from 98.6% and 97.9% (with GA) to 99.7% (with CFS), respectively. These findings validate the efficacy of the CFS method in improving overall classification accuracy.

TABLE VI. Accuracy Results

Classifier	With GA	With CFS
DT	0.993	0.999
RF	0.993	0.999
SVM	0.986	0.997
CNN	0.979	0.997

b) Precision: Precision assesses the ratio of accurate positive identifications. As illustrated in Table VII and Fig. 5, DT and RF achieved the highest precision rate of 99.9% when utilizing CFS. SVM and CNN also exhibited enhancements, increasing to 99.5% with CFS, in contrast to their previous rates of 98.5% and 98.6% with GA. This indicates that the models improved in their ability to reduce false positives when employing CFS.

c) Recall: Recall reflects the model's capacity to identify all genuine instances of attacks accurately. As shown in Table VIII and Fig. 6, every classifier attained a recall rate of 99.9%, except for the CNN model utilizing GA-selected

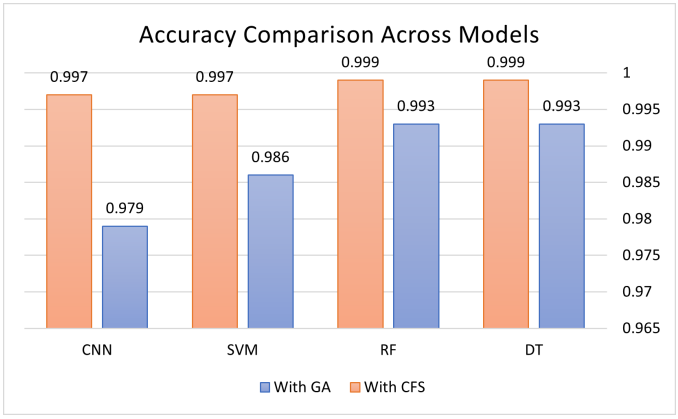


Fig. 4. Accuracy results.

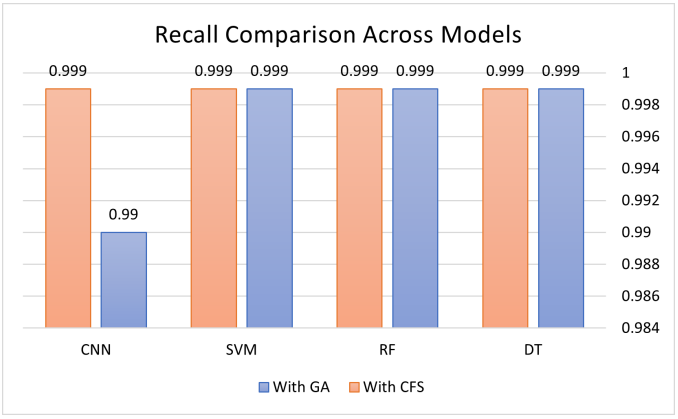


Fig. 6. Recall results.

TABLE VII. PRECISION RESULTS

Classifier	With GA	With CFS
DT	0.993	0.999
RF	0.993	0.999
SVM	0.985	0.995
CNN	0.986	0.995

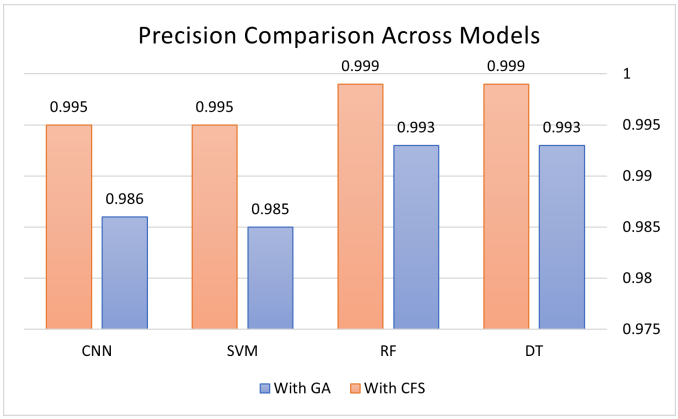


Fig. 5. Precision results.

features, which experienced a minor decrease to 99.0%. The consistently elevated recall values indicate that the models are exceptionally proficient in recognizing actual attacks.

TABLE VIII. RECALL RESULTS

Classifier	With GA	With CFS
DT	0.999	0.999
RF	0.999	0.999
SVM	0.999	0.999
CNN	0.990	0.999

d) F1 Score: The F1-score represents the harmonic mean of precision and recall, establishing a balance between these two metrics. As demonstrated in Table IX and Fig. 7 DT and RF achieved the highest F1-score of 99.9% when utilizing CFS, whereas SVM and CNN attained scores of 99.7%, an increase from 99.2% and 98.8% with GA, respectively. These findings validate the robustness of the models, especially when employing CFS.

TABLE IX. F1 SCORE RESULTS

Classifier	With GA	With CFS
DT	0.996	0.999
RF	0.996	0.999
SVM	0.992	0.997
CNN	0.988	0.997

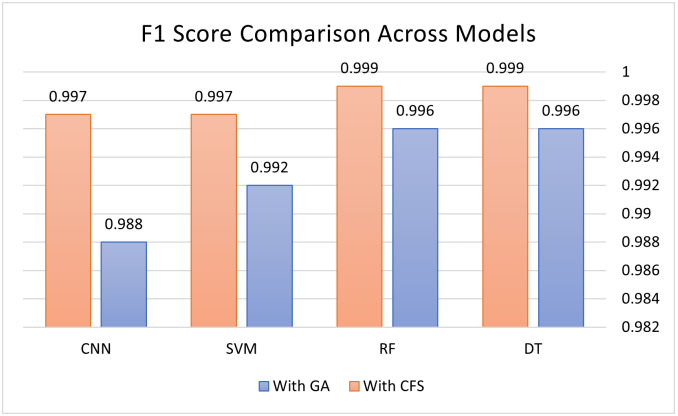


Fig. 7. F1 score.

VII. DISCUSSION

This section interprets the experimental results, highlights the key findings, and compares our approach with existing studies to provide deeper insights into the significance of the work.

A. Key Findings

According to the comprehensive results derived from our experiments, the classifiers that exhibited the highest performance in terms of accuracy, precision, recall, and F1-score were DT and RF, particularly when trained with features selected through CFS. Both models attained an impressive 99.9% across all metrics, underscoring their exceptional reliability and effectiveness in identifying malicious activities within IoT networks.

In terms of computational efficiency, DT surpassed all other classifiers, achieving the quickest training and testing durations with both GA and CFS. This outcome affirms that DT is particularly well-suited for real-time applications and scenarios that necessitate rapid processing with minimal computational overhead.

While the CNN and SVM models demonstrated commendable detection accuracy when CFS was utilized, they exhibited longer training and testing durations, particularly with GA. Furthermore, SVM with GA selection recorded the lowest precision, likely attributable to the algorithm's sensitivity to high-dimensional data and correlated features, which can adversely affect its robustness when handling larger or imbalanced datasets.

Overall, the CFS feature selection method has proven to be more efficient and stable across all classifiers, delivering superior performance compared to GA in most instances. This highlights the critical role of appropriate feature selection in improving classifier accuracy and minimizing computational expenses.

B. Comparative Analysis between Our Proposed Approach and an Existing Study

This section provides a comparative analysis of our proposed ML based IoT security framework in relation to the approach outlined by Nazir et al. [6]. While both studies focus on improving IoT security using ML techniques, they exhibit notable differences in their methodologies, system architectures, and performance results.

Nazir et al. [6] proposed a full-stack collaborative threat intelligence framework that integrates blockchain, ML, and human feedback via an iOS control center. This approach emphasizes real-time, collaborative detection and mitigation, utilizing advanced models like RF, DT, Long LSTM, and CNN. Their study focused on reducing false negatives and improving adaptability in IoT security, primarily using the IoT23 dataset for model training and evaluation. In contrast, our approach is a more streamlined ML-based anomaly detection framework that relies on feature selection to improve model efficiency without requiring additional infrastructure like blockchain. We used four classifiers (DT, RF, SVM, and CNN) and employed two feature selection methods—GA and CFS to enhance detection accuracy and reduce computational overhead.

Our results showed a clear advantage in detection accuracy compared to Nazir et al.'s approach. As illustrated in Fig. 8, our DT and RF models achieved 99.9% accuracy, precision, recall, and F1-score when using CFS, outperforming the 95.0% accuracy reported for the best-performing model (RF) in Nazir et al.'s study. Similarly, our SVM and CNN models reached 99.7% accuracy with CFS, significantly higher than the 92.0% achieved by the CNN model in their study. This marked improvement highlights the effectiveness of our feature selection strategy in boosting model performance, even for DL classifiers like CNN.

Optimizing computational efficiency is essential for real-time IoT applications, and our research revealed notable benefits in this aspect. Our DT model achieved the quickest training and testing durations, finishing training in 0.6821 seconds (GA) and 1.1308 seconds (CFS), with testing times of 0.0168 seconds (GA) and 0.0285 seconds (CFS). In contrast, Nazir et al. did not provide specific training and testing durations, but their use of blockchain and collaborative methods indicates a more resource-demanding strategy. Additionally, our RF model demonstrated effective processing, with training times of 33.4947 seconds (GA) and 43.9254 seconds (CFS), and testing times of 1.1156 seconds (GA) and 1.0211 seconds (CFS). This performance is considerably quicker than what one might anticipate from a comprehensive system like that of Nazir et al., which includes blockchain overhead and real-time feedback loops. On the other hand, our SVM model showed a distinct trade-off, needing 5448.912 seconds (GA) and 158.8664 seconds (CFS) for training, and 1005.804 seconds (GA) and 94.2506 seconds (CFS) for testing. This underscores the computational demands of SVM, despite its impressive accuracy, making it less feasible for real-time applications without additional optimization. The CNN, while also achieving high accuracy, required moderate training and testing times, clocking in at 188.9341 seconds (GA) and 387.9729 seconds (CFS) for training, and 6.1233 seconds (GA) and 24.252 seconds (CFS) for testing.

Although both studies used the IoT23 dataset, our method showed better generalization, consistently achieving high performance across all metrics. This is especially important for real-world IoT settings, where accurately identifying anomalies from various attack types is crucial. While Nazir et al.'s framework is innovative, it might encounter scalability issues due to the extra overhead from its collaborative and blockchain features.

Our research emphasizes the significance of effective feature selection in lowering computational costs and enhancing detection accuracy. The notable performance improvements seen with CFS indicate that future research should focus on this method, particularly in resource-limited IoT environments. Furthermore, our framework's efficient design, which does not depend on external infrastructure, provides a more feasible solution for real-time IoT security.

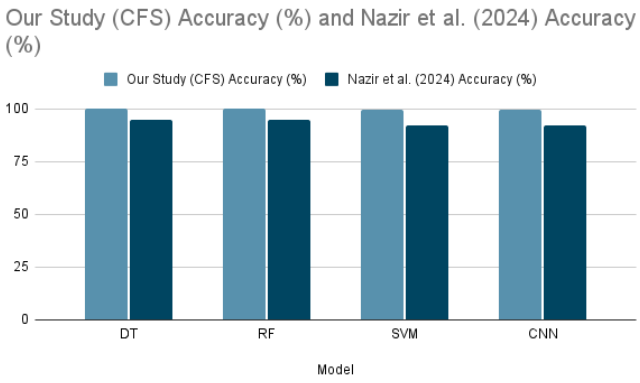


Fig. 8. Accuracy comparison between this study and Nazir et al. for different ML models.

VIII. Conclusion and Future Work

In this paper, we proposed a robust framework to enhance the security of IoT environments by leveraging ML-based

anomaly detection techniques. The framework was designed to address key security challenges in IoT networks, particularly those related to detecting sophisticated attacks such as botnets and DoS. We implemented four widely used classifiers—DT, RF, SVM, and CNN on the IoT23 dataset, a comprehensive resource designed for IoT threat detection.

To further optimize the performance of our classifiers, we incorporated two feature selection algorithms: GA and CFS. The comparative evaluation of these algorithms revealed that CFS consistently outperformed GA in terms of classification accuracy and execution speed. This confirmed the importance of effective feature selection in reducing model complexity and improving detection reliability.

Our experimental results showed that both DT and RF classifiers achieved exceptional performance across all metrics—accuracy, precision, recall, and F1-score—reaching 99.9% when paired with CFS. Moreover, DT exhibited the lowest training and testing times, making it the most suitable model for real-time and resource-constrained IoT environments. While CNN and SVM also showed strong performance, they were more computationally intensive, especially when used with GA.

Despite these promising results, there are some limitations to consider. The study relies on the IoT23 dataset, which may not encompass all possible IoT attack scenarios, and model performance could vary in different network environments or device configurations. Additionally, potential biases in the training data could affect detection accuracy. Addressing these limitations in future work will further strengthen the generalizability and reliability of the proposed framework.

In conclusion, our contribution lies in designing and validating a machine learning-based anomaly detection framework that is both accurate and computationally efficient. Future work will focus on extending the framework to support additional datasets, addressing adversarial threats, and enabling dynamic adaptation in evolving network environments.

For future work, we will conduct and evaluate the performance of our proposed model using other datasets, such as the IoTID20 dataset, as it is recommended for the same condition under which we focus. Moreover, we will try to test it again by selecting other selection algorithms and try to select fewer features with new attack types collected from real-world IoT environments. Evaluating and implementing other types of AI algorithms, such as ML classifiers, will add value for future projects.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [GRANT 252792]. The authors would like to thank the anonymous reviewers for their insightful scholastic comments and suggestions, which improved the quality and clarity of the paper.

REFERENCES

[1] S. M. Tahsien, H. Karimipour, and P. Spachos, "Machine learning based solutions for security of internet of things (iot): A survey," *Journal of Network and Computer Applications*, vol. 161, p. 102630, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804520301041

[2] T. Mazhar, D. B. Talpur, T. A. Shloul, Y. Y. Ghadi, I. Haq, I. Ullah, K. Ouahada, and H. Hamam, "Analysis of iot security challenges and its solutions using artificial intelligence," *Brain Sciences*, vol. 13, no. 4, 2023. [Online]. Available: https://www.mdpi.com/2076-3425/13/4/683

[3] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for internet of things (iot) security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020.

[4] R. Ahmad and I. Alsmadi, "Machine learning approaches to iot security: A systematic literature review[formula presented]," 6 2021.

[5] M. A. Amanullah, R. A. Ariyaluran Habeeb, F. Nasaruddin, A. Gani, E. Ahmed, A. Nainar, N. Akim, and M. Imran, "Deep learning and big data technologies for iot security," *Computer Communications*, vol. 151, 02 2020.

[6] A. Nazir, J. He, N. Zhu, A. Wajahat, F. Ullah, S. Qureshi, X. Ma, and M. S. Pathan, "Collaborative threat intelligence: Enhancing iot security through blockchain and machine learning integration," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, 2 2024.

[7] K. Kiran, R. Devisetty, N. Kalyan, K. Mukundini, and K. Ramesh, "Building a intrusion detection system for iot environment using machine learning techniques," *Procedia Computer Science*, vol. 171, pp. 2372–2379, 01 2020.

[8] F. Alwahedi, A. Aldhaheri, M. A. Ferrag, A. Battah, and N. Tihanyi, "Machine learning techniques for iot security: Current research and future vision with generative ai and large language models," *Internet of Things and Cyber-Physical Systems*, 01 2024.

[9] F. H. ZAWAIDEH and S. Y. A. BAKER, "Enhanced iot security using cascaded machine learning," *International Journal of Research Studies in Computer Science and Engineering*, vol. 9, pp. 23–34, 2023.

[10] M. Bagaa, T. Taleb, J. B. Bernabe, and A. Skarmeta, "A machine learning security framework for iot systems," *IEEE Access*, vol. 8, pp. 114 066–114 077, 2020.

[11] A. Boukerche and R. W. Coutinho, "Design guidelines for machine learning-based cybersecurity in internet of things," *IEEE Network*, vol. 35, pp. 393–399, 3 2021.

[12] A. Makkar, S. Garg, N. Kumar, M. S. Hossain, A. Ghoneim, and M. Alrashoud, "An efficient spam detection technique for iot devices using machine learning," *IEEE Transactions on Industrial Informatics*, vol. PP, pp. 1–1, 01 2020.

[13] J. Dhanke, D. Kamalraj, G. Ramesh, K. Sankaran, S. Sharma, and D. Khasim, "A machine learning based iot for providing an intrusion detection system for security," *Microprocessors and Microsystems*, vol. 82, p. 103741, 01 2021.

[14] A. R. Gad, M. Haggag, A. A. Nashat, and T. M. Barakat, "A distributed intrusion detection system using machine learning for iot based on ton-iot dataset," *International Journal of Advanced Computer Science and Applications*, vol. 13, pp. 548–563, 2022.

[15] F. Alrowais, S. Althahabi, S. Alotaibi, A. Mohamed, A. Hamza, R. Marzouk, and M. Ahmed, "Automated machine learning enabled cybersecurity threat detection in internet of things environment," *Computer Systems Science and Engineering*, vol. 45, pp. 687–700, 08 2022.

[16] A. Awajan, "A novel deep learning-based intrusion detection system for iot networks," *Computers*, vol. 12, 2 2023.

[17] U. Tahir, M. Abid, M. Fuzail, and N. Aslam, "Enhancing iot security through machine learning-driven anomaly detection," *VFAST Transactions on Software Engineering*, vol. 12, pp. 01–13, 05 2024.

[18] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, "Intrusion detection system using feature extraction with machine learning algorithms in iot," *Journal of Sensor and Actuator Networks*, vol. 12, 4 2023.

[19] M. A. Khatun, S. F. Memon, C. Eising, and L. L. Dhirani, "Machine learning for healthcare-iot security: A review and risk mitigation," *IEEE Access*, vol. 11, pp. 145 869–145 896, 2023.

[20] A. Churcher, R. Ullah, J. Ahmad, S. U. Rehman, F. Masood, M. Gogate, F. Alqahtani, B. Nour, and W. J. Buchanan, "An experimental analysis of attack classification using machine learning in iot networks," *Sensors (Switzerland)*, vol. 21, pp. 1–32, 1 2021.

[21] I. Mukherjee, N. K. Sahu, and S. K. Sahana, "Simulation and modeling for anomaly detection in iot network using machine learning," *International journal of wireless information networks*, vol. 30, no. 2, pp. 173–189, 2023.

[22] P. Bedi, S. Mewada, R. A. Vatti, C. Singh, K. S. Dhindsa, M. Ponnusamy, and R. Sikarwar, "Retracted: Detection of attacks in iot sensors networks using machine learning algorithm," *Microprocessors and Microsystems*, vol. 82, p. 103814, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141933120309595

[23] E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, "Anomaly detection ids for detecting dos attacks in iot networks based on machine learning algorithms," *Sensors*, vol. 24, no. 2, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/2/713

[24] B. Alotaibi and M. Alotaibi, "A stacked deep learning approach for iot cyberattack detection," *Journal of Sensors*, vol. 2020, no. 1, p. 8828591, 2020.

[25] A. Huč and D. Trček, "Anomaly detection in iot networks: From architectures to machine learning transparency," *IEEE Access*, vol. 9, pp. 60 607–60 616, 2021.

[26] E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, "Anomaly detection ids for detecting dos attacks in iot networks based on machine learning algorithms," *Sensors*, vol. 24, no. 2, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/2/713

DAE-IDS: A Domain-Aware Ensemble Intrusion Detection System with Explainable AI for Industrial IoT Networks

Saifur Rahman
Electrical Engineering Department-College of Engineering, Najran University, Najran 61441, Saudi Arabia

***Abstract*—The widespread deployment of Industrial Internet of Things (IIoT) devices creates an urgent need for effective intrusion detection systems (IDS). However, two critical challenges limit current approaches: severe class imbalance in network traffic data that hampers detection of rare attacks, and the "black-box" nature of machine learning models that undermines trust in security-critical applications. This study presents a Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) equipped with explainable AI, addressing both challenges through frequency-aware ensemble learning and computationally efficient interpretability mechanisms. Using the Edge-IIoTset dataset containing 80 features across 12 classes, attacks were categorized into three frequency groups: majority attacks (5 classes), middle-frequency attacks (4 classes), and minority attacks (3 classes). Specialized Random Forest models (50 trees each with class weighting) tailored to each frequency group, then developed a domain-aware ensemble that routes traffic to the most appropriate specialized model based on attack frequency patterns. To enhance interpretability, SHAP explanations added using an optimized approach that combines interventional TreeExplainer with instance subsampling (300 samples per model) and top-k feature prioritization. This optimization reduced SHAP computation time by 60% while maintaining full interpretability. The domain-aware ensemble achieved superior performance with a macro-F1 score of 1.00, demonstrating significant improvements in rare-attack detection compared to traditional approaches. SHAP analysis revealed attack-specific discriminative features, providing actionable insights for security analysts. This framework successfully bridges the accuracy-interpretability trade-off in IIoT security applications, enabling trustworthy intrusion detection suitable for resource-constrained edge environments. The attack-frequency specialization approach offers a practical solution for handling class imbalance while maintaining model transparency through efficient explainability mechanisms.**

Keywords*—Intrusion detection systems; IoT security; Explainable AI (XAI); class imbalance; frequency-aware ensemble; SHAP interpretability; domain-aware routing; confidence-based ensemble; Edge-IIoTset dataset; optimized random forest

I. Introduction

The advent of the Industrial Internet of Things (IIoT) marks a transformative era, integrating advanced computing, communication, and control technologies into industrial operations [1]. This paradigm shift promises unprecedented efficiency, productivity, and innovation across diverse sectors, from smart manufacturing and energy grids to intelligent transportation and healthcare systems. By connecting physical devices, sensors, and actuators with cyber systems, IIoT facilitates real-time data exchange, remote monitoring, and automated decision-making, thereby optimizing complex industrial processes [2].

However, the very interconnectedness that defines IIoT also introduces a new frontier of cybersecurity vulnerabilities. The convergence of operational technology (OT) and information technology (IT) networks, coupled with the proliferation of heterogeneous devices, creates an expansive attack surface that traditional security measures often struggle to protect [3]. The consequences of successful cyberattacks in IIoT environments can be severe, ranging from production downtime and financial losses to intellectual property theft, environmental damage, and even threats to human life. Therefore, developing robust and adaptive cybersecurity solutions, particularly effective intrusion detection systems (IDS), is paramount to safeguarding the integrity, availability, and confidentiality of IIoT infrastructure [4].

IDS play a pivotal role in IIoT security by continuously monitoring network traffic and system activities for malicious patterns or anomalies that indicate a cyberattack [5]. Given the unique characteristics of IIoT environments—such as resource-constrained devices, real-time operational demands, and the criticality of physical processes—IDS solutions must be highly efficient, accurate, and resilient. The evolving landscape of cyber threats, including sophisticated malware, distributed denial-of-service (DDoS) attacks, and advanced persistent threats (APTs), necessitates constant innovation in IDS methodologies [6].

In recent years, the research community has advanced IIoT security through machine learning (ML) and deep learning (DL) techniques, novel datasets like Edge-IIoTset, and hybrid models addressing data imbalance and efficiency. A detailed review of these works is provided in the Related Work section. Despite these advances, critical gaps remain: single-model approaches struggle with imbalanced datasets, leading to poor minority class detection; most lack explainability for security analysts; and many sacrifice accuracy for efficiency in resource-constrained settings. Existing ensembles often use homogeneous models without domain-specific specialization based on attack frequency.

To address these research gaps, this paper presents a novel Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) with the following key contributions:

1) Domain-aware ensemble architecture: A three-model Random Forest ensemble trained on the Edge-IIoT dataset, partitioned into high-frequency, medium-frequency, and low-frequency attack classes for specialized detection.

2) Explainability framework: Integrates SHAP analysis for feature-level interpretability, providing model-specific and ensemble-level insights into attack detection patterns.

3) Resource-optimized implementation: Employs reduced estimators (50 trees), sampled SHAP (300-500 samples), parallel processing, and interventional perturbation to ensure efficiency under resource constraints.

4) Empirical validation: Evaluated on the Edge-IIoT dataset, DAE-IDS achieves high accuracy, handles class imbalance effectively, and outperforms traditional single-model approaches in interpretability and efficiency.

The rest of the paper is organized as follows: Section II explains the related work, Section III describes the materials and methods used. Section IV presents a detailed description of the results. Section V concludes the paper.

II. Related Work

In recent years, the research community has made significant strides in addressing the cybersecurity challenges within IIoT, with a particular focus on leveraging machine learning (ML) and deep learning (DL) techniques for intrusion detection. Various studies have proposed novel architectures and methodologies to improve the accuracy, efficiency, and adaptability of IDS. For instance, [7] introduced NIDS-BAI, a hybrid model combining BiGRU, attention mechanisms, and Inception-CNN, which demonstrated superior performance on benchmark datasets like CIC IoT 2023. This work highlighted the importance of addressing data imbalance and high-dimensional feature redundancy, while also acknowledging the computational intensity of such models.

To facilitate more realistic evaluations, the Edge-IIoTset dataset was developed by [8], providing a comprehensive testbed that integrates real devices and attacks, proving invaluable for deep learning model assessment. Building on this, [9] evaluated the efficacy of DenseNet and Inception Time models on various datasets, underscoring Inception Time's capability in multiclass classification with minimal computational overhead, a crucial factor for IIoT deployments. In [10], authors propose a deep learning-based anomaly detection system using Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU) to secure IIoT environments from cyber threats, demonstrating superior performance on the Edge-IIoTset dataset. Similarly, [11] empirically assessed the Edge-IIoT-2022 dataset, observing high accuracy for binary classification but noting performance reductions in multiclass scenarios due to data imbalance, which remains a persistent challenge in IDS development.

Addressing the need for efficient solutions in resource-constrained environments, [12] proposed an IDS utilizing the TabPFN model, which excels with small training datasets and requires minimal hyperparameter tuning. Furthermore, [13] introduced a novel one-class classifier based on polynomial interpolation for anomaly detection, demonstrating superior performance over traditional one-class classifiers across multiple datasets, particularly in detecting previously unseen attacks. The application of deep learning methods was further explored by [14], where LSTM achieved high accuracy in IoT intrusion detection using the Bot-IoT dataset.

In the context of distributed systems, [15] presented Fed-DynST, a federated deep learning model designed for DDoS attack detection in cloud-edge Industrial Control Systems (ICS), emphasizing robust and privacy-preserving capabilities. Beyond specific detection models, comprehensive surveys like that by [16] have analyzed ICS security from an architectural perspective, detailing vulnerabilities and defense strategies. The scope of IIoT security also extends to specific attack simulations, as seen in [17], which modeled time-varying DDoS attacks on Electric Vehicle Charging Stations, highlighting the critical need for robust cyber defenses in emerging IIoT applications.

Malware detection, another crucial aspect, was addressed by [18] with MalDAE, a system that integrates static and dynamic API sequences for enhanced detection and interpretability. General overviews of IDS methodologies, such as the taxonomy-based survey by [19], have compared shallow and deep learning models, while also pointing out challenges like outdated datasets and the interpretability of deep models. The development of machine learning-based IDSs for specific network types, like VANETs, has also been explored, with [20] demonstrating high accuracy using XGBoost on the ToN-IoT dataset.

Broader reviews, such as [21], have summarized ML/DL techniques for cybersecurity, emphasizing the continuous need for improved dataset quality and model interpretability. Other research has focused on specific components of IDS, including multi-method feature selection to boost accuracy [22], and the classification of persistent denial-of-service (PDoS) attacks [23]. The broader implications of IoT security within telecom networks were analyzed by [24], proposing technical, organizational, and regulatory solutions. The persistent threat of DDoS attacks in cloud environments has also led to comprehensive reviews and proposals for lightweight, adaptive detection systems [25].

Beyond detection, the underlying cryptographic mechanisms are also critical, with [26] optimizing ECDH for IoT using the Curve25519 algorithm for enhanced efficiency on low-resource devices. Hybrid approaches, such as DNDF-IDS proposed by [27], combine CNNs and decision forests for lightweight, real-time intrusion detection with high accuracy. Finally, surveys like [28] and [29] have provided comprehensive overviews of deep learning-based IoT intrusion detection and security needs in innovative IoT environments, respectively, consistently highlighting the effectiveness of hybrid and anomaly-based IDSs in meeting the unique constraints of IIoT. This body of work underscores the dynamic nature of IIoT security research and the ongoing efforts to develop more sophisticated and resilient intrusion detection capabilities.

III. Materials and Methods

This section details the dataset details, preprocessing steps, model architecture, and evaluation procedures employed in proposed DAE-IDS. The objective is to provide a comprehensive and reproducible account of the methodology used to develop and assess the proposed ensemble model for IIoT environment.

A. Dataset Description

The dataset utilized in this study is the Edge-IIoTset, a contemporary dataset specifically designed for intrusion detection in IIoT environments. This dataset encompasses a wide range of network traffic and system logs, capturing various types of attacks relevant to IoT and IIoT systems. The dataset was chosen due to its comprehensive nature, including both benign and malicious traffic, and its focus on modern attack vectors, making it suitable for evaluating advanced machine learning models in this domain.

Upon initial loading, the dataset contained approximately 1.9 million records and 48 features. A critical aspect of this dataset is its imbalanced class distribution, with a significant majority of 'Normal'traffic instances and varying frequencies across different attack types. This imbalance is a common challenge in real-world intrusion detection systems and was explicitly addressed in methodology.

B. Data Preprocessing

Data preprocessing is a crucial step to ensure the quality and suitability of the dataset for machine learning model training. The raw Edge-IIoTset dataset underwent several preprocessing stages, including data cleaning, feature selection, and normalization.

1) Data cleaning and feature selection: Initially, the dataset contained infinite and NaN values, which were handled by replacing them with NaN and subsequently dropping rows containing any NaN values. This step ensures that the models are trained on clean and valid data. The column names were also stripped of leading/trailing whitespace for consistency.

Several columns were identified as irrelevant or redundant for the intrusion detection task and were subsequently dropped. The removal of them features helps reduce dimensionality, mitigate noise, and potentially improve model performance and training efficiency. Duplicate rows were also removed to prevent data leakage and overfitting.

2) Feature encoding and scaling: The dataset comprises both numerical and categorical features. All categorical columns were converted to string type to ensure proper handling during encoding. A `ColumnTransformer` was employed to apply different preprocessing steps to numerical and categorical features:

a) Numerical features: These features were scaled using `MinMaxScaler`. This technique scales features to a given range (typically 0 to 1), which is essential for algorithms sensitive to feature scales, such as those used in neural networks and distance-based methods.

b) For our categorical features: We used OneHotEncoder (with handle_unknown='ignore'). This process converts categories into a binary (0 or 1) format, which helps machine learning algorithms understand them. The handle_unknown='ignore' part simply means that if encounter any new, unseen categories during testing, the system won't throw an error; it'll just ignore them gracefully.

After transformation, the preprocessed features were converted into a Pandas DataFrame, and new feature names were generated to reflect the one-hot encoded categorical features. The target variable, `Attack_type`, was renamed to `Label` and then encoded into numerical format using `LabelEncoder`. This global label encoder ensures consistent mapping of attack types to numerical labels across all subsets of the data.

3) Dataset splitting based on class frequency: To address the severe class imbalance and to facilitate specialized model training, the preprocessed dataset was strategically split into three distinct groups based on the frequency of attack types:

a) Majority classes: This group includes high-frequency attack types (e.g. Normal, DDoS_UDP, DDoS_ICMP, DDoS_HTTP, DDoS_TCP) that constitute a significant portion of the dataset.

b) Middle classes: This group comprises medium-frequency attack types (e.g. Vulnerability_scanner, Backdoor, Port_Scanning, Ransomware).

c) Minority classes: This group consists of low-frequency attack types (e.g. Fingerprinting, MITM, XSS).

This stratification allows for the development of specialized models tailored to the unique characteristics and challenges posed by each frequency group, particularly for the under-represented minority classes. Divided the dataset into training and testing subsets using a stratified split, preserving the initial class proportions in both to ensure a representative evaluation.

C. Model Architecture and Ensemble Strategies

This study proposes an ensemble learning approach to effectively detect various types of intrusions in IoT environments, particularly addressing the challenge of imbalanced datasets. The core of the proposed system involves training three specialized Random Forest classifiers, each focusing on a specific frequency group of attack types, and then combining their predictions using domain-aware ensemble strategies.

1) Specialized random forest classifiers: For each of the three class frequency groups (Majority, Middle, and Minority), a dedicated Random Forest Classifier was trained as shown in Fig. 1. Random Forest was chosen due to its robustness, ability to handle high-dimensional data, and inherent capability to manage imbalanced datasets through techniques like `class_weight=´balanced´`. Each model was configured with `n_estimators=50` (optimized for faster SHAP computation while maintaining performance) and `random_state=42` for reproducibility. Parallel processing (`n_jobs=-1`) was enabled to accelerate training.

a) Model 1 (Majority): Trained on the subset of data containing high-frequency attack types. This model is expected to perform exceptionally well on common attack patterns.

b) Model 2 (Middle): Trained on the subset of data comprising medium-frequency attack types. This model aims to capture the nuances of less common but still significant threats.

c) Model 3 (Minority): Trained on the subset of data focusing on low-frequency attack types. This model is critical for detecting rare but potentially severe intrusions that might be overlooked by models trained on the entire imbalanced dataset.

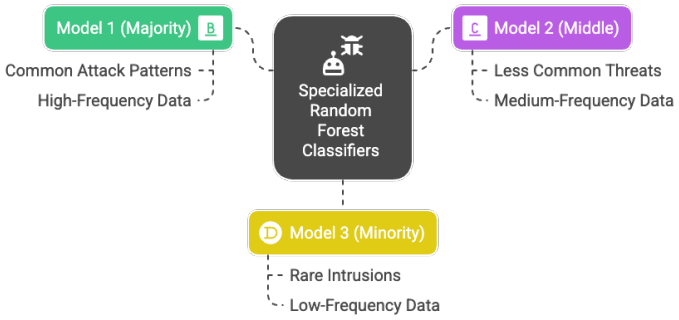


Fig. 1. Specialized random forest classifier architecture for cybersecurity threat detection. Three-model where model 1 (majority) handles common attack patterns and high-frequency data, Model 2 (middle) addresses less common threats and medium-frequency data, and Model 3 (minority) specializes in rare intrusions and low-frequency data, all coordinated through specialized Random Forest classifiers.

2) Domain-aware ensemble: The domain-aware ensemble strategy (Fig. 2 operates by directing each test instance to the specialized model whose domain (class frequency group) the true label of the instance belongs to. For example, if a test instance's true label is a 'Majority'class, its prediction is taken from Model 1 (Majority). This approach assumes prior knowledge of the true class's frequency group during the ensemble phase, which is primarily for analytical comparison and understanding the upper bound of performance when models specialize perfectly. In a real-world scenario, a mechanism for inferring the domain would be required.

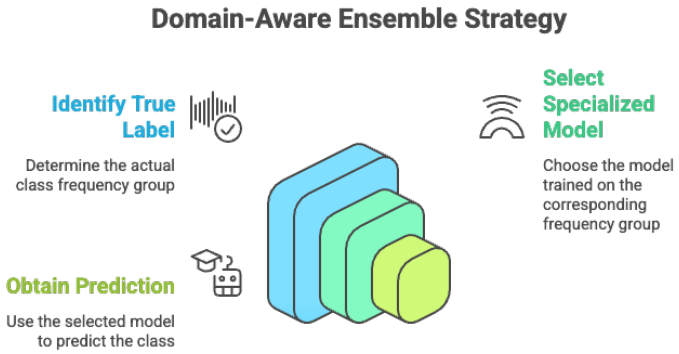


Fig. 2. Domain-aware ensemble strategy for frequency-based model selection. Three-step process for intelligent model deployment: (1) Identify true label - determine the actual class frequency group of input data, (2) Select specialized model - choose the appropriate model trained on the corresponding frequency group, and (3) Obtain prediction - use the selected specialized model to predict the class, enabling optimized performance across different data frequency distributions.

D. Evaluation Metrics and Interpretability Analysis

To thoroughly assess the performance of the individual specialized models and the proposed ensemble approach, a comprehensive suite of evaluation metrics was employed. Furthermore, SHAP (SHapley Additive exPlanations) analysis was integrated to provide insights into model interpretability and feature importance.

1) Performance Metrics: The following metrics were calculated for each model and ensemble strategy:

- Accuracy: The ratio of correctly predicted instances to the total number of instances.
- Balanced Accuracy: The arithmetic mean of recall scores across all classes, designed to evaluate classifiers on imbalanced datasets.
- Precision (Macro/Micro/Weighted): The fraction of true positives among all positive predictions, computed globally (Micro), as a class-wise unweighted mean (Macro), or class-wise mean weighted by support (Weighted).
- Recall (Macro/Micro/Weighted): The proportion of true positives relative to actual positive instances, aggregated across classes via global (Micro), unweighted (Macro), or support-weighted (Weighted) averaging.
- F1-Score (Macro/Micro/Weighted): The harmonic mean of precision and recall, providing a balanced performance measure with identical aggregation schemes as precision and recall.
- Cohen's Kappa: A statistic quantifying inter-annotator agreement for categorical items, adjusted for chance-level agreement.
- Matthews Correlation Coefficient (MCC): A contingency-matrix-based measure of classification quality that is robust to class size imbalance in both binary and multi-class settings.
- ROC-AUC (OvR/OvO): The area under the Receiver Operating Characteristic curve, computed via One-vs-Rest (OvR) or One-vs-One (OvO) strategies for multi-class problems, reflecting the model's class-discrimination capability.
- Log Loss: A performance metric for probabilistic classifiers that penalizes divergence between predicted probabilities and true labels, defined as the negative log-likelihood of the model.
- Average Precision (AP): The weighted mean of precision values at all classification thresholds, with weights corresponding to recall increments from preceding thresholds

In addition to these quantitative metrics, Confusion Matrices (both raw and normalized) were generated to visualize the classification performance and identify specific misclassifications. ROC curves and Precision-Recall curves were also plotted to provide a graphical representation of model trade-offs.

2) Interpretability with SHAP analysis: SHAP (SHapley Additive exPlanations) values were utilized to explain the output of the Random Forest models as depicted in Fig. 3. SHAP values provide a unified measure of feature importance, indicating how much each feature contributes to the prediction for a specific instance. This allows for both global and local interpretability of the models.

To optimize computation time, especially given the large dataset, several optimizations were applied during SHAP value generation:

- Reduced SHAP sample size: SHAP analysis was performed on a smaller, representative sample of the test data (e.g. 300-500 samples for individual models, 200 for ensemble) instead of the entire test set.
- TreeExplainer with interventional perturbation: `shap.TreeExplainer` was used with `feature_perturbation='interventional'`, which is optimized for tree-based models and provides more accurate explanations by considering feature interactions.
- Limited summary plots Summary plots were limited to display only the top 15-20 most important features, focusing on the most influential factors.
- Optimized ensemble SHAP computation For ensemble models, SHAP values were collected efficiently by applying the appropriate specialized model's explainer to each sample based on its true label (for domain-aware ensemble) or the highest confidence model (for confidence-based ensemble).

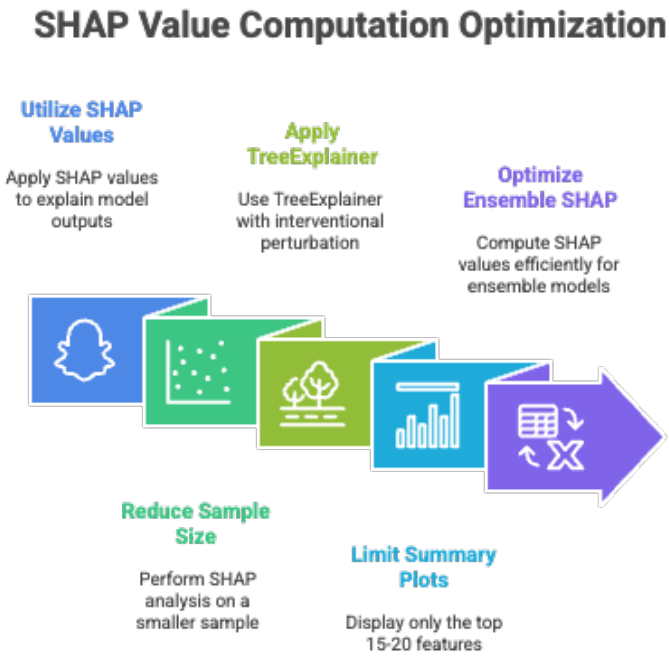


Fig. 3. SHAP value computation optimization workflow. Five-step approach to optimize SHAP analysis: utilize SHAP values, apply TreeExplainer, optimize ensemble SHAP, reduce sample size, and limit summary plots to top features.

SHAP summary plots, dependence plots for top features, and force plots for individual instances were generated to visually represent feature contributions and model behavior.

IV. Results

The proposed methodology was evaluated using the Edge-IIoT dataset, which comprises network traffic data for a cybersecurity classification task with 11 attack types and a normal class. All experiments were conducted in a Python environment utilizing standard machine learning libraries such as `scikit-learn`, `pandas`, `numpy`, `matplotlib`, `seaborn`, and `shap`. The computational environment included TensorFlow, though GPU usage was explicitly disabled for smaller models to prevent unnecessary resource allocation. Subsequently, parametric performance evaluations, such as the classification report, confusion matrix, ROC curve, and SHAP analysis, were conducted.

A. Comprehensive Matrices of All Models

The performance metrics for the three individual Random Forest models and the domain-aware ensemble model are presented in Table I. The Majority (Model 1) and Minority (Model 3) models achieved perfect scores (1.0000) across all metrics, including accuracy, balanced accuracy, precision, recall, F1-score, Cohen's Kappa, Matthews Correlation Coefficient (MCC), and ROC-AUC (both One-vs-Rest and One-vs-One). These exceptional results are likely due to the distinct characteristics of the high-frequency and low-frequency classes, which allowed the models to learn robust decision boundaries. The extremely low log loss values (0.0024 for Model 1 and 0.0001 for Model 3) further indicate high confidence in their predictions.

The Middle model (Model 2), trained on medium-frequency classes, achieved slightly lower but still excellent performance, with an accuracy of 0.9846, macro-average F1-score of 0.9736, and MCC of 0.9770. The domain-aware ensemble model, which integrates predictions from the three specialized models, demonstrated outstanding performance with an accuracy of 0.9994, macro-average F1-score of 0.9950, and MCC of 0.9986. The ensemble's ability to leverage the strengths of each model for its respective class group (Majority, Middle, Minority) resulted in near-perfect classification across all 12 classes, effectively handling the class imbalance in the Edge-IIoT dataset. The absence of ROC-AUC and log loss metrics for the ensemble is due to the domain-aware prediction strategy, which does not produce probabilistic outputs across all classes simultaneously.

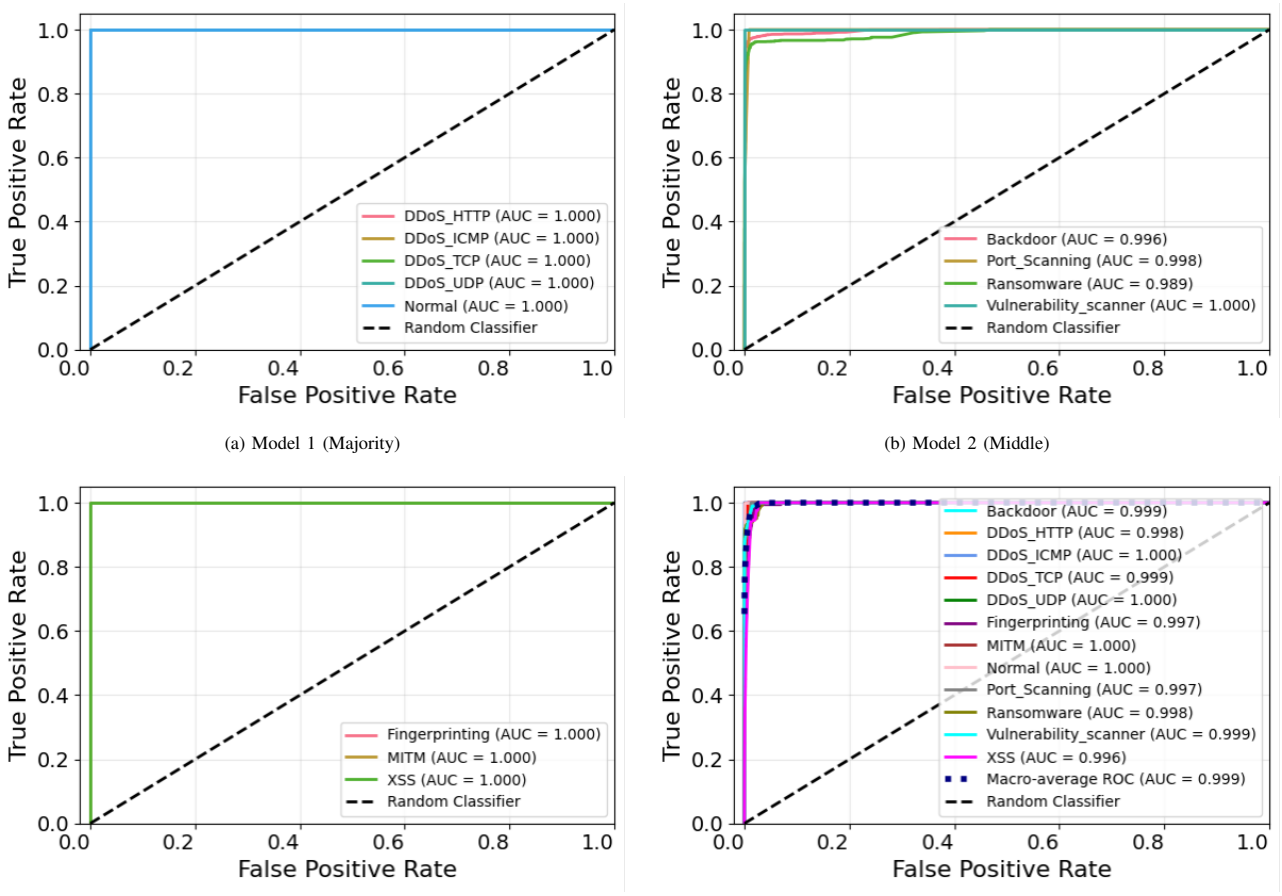
B. ROC Curve Analysis

The performance of the proposed models was further evaluated using Receiver Operating Characteristic (ROC) curves, which plot the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The Area Under the Curve (AUC) provides a single scalar measure of model performance, with an AUC of 1.0 indicating perfect classification and 0.5 representing a random classifier. The ROC curves for each model are presented below (Fig. 4, derived from the Edge-IIoT dataset with 12 classes, and analyzed based on the specialized training and ensemble strategy.

The ROC curve for Model 1, trained on five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal), achieves an AUC of 1.000 for each class. The curve reaches a true positive rate (TPR) of 1.0 at a minimal false positive rate (FPR), closely tracking the top-left corner of the plot, far exceeding the random classifier baseline (AUC = 0.5). High support values (e.g., Normal: 276,172, DDoS_UDP: 24,313) and distinct feature distributions likely contribute to the Random Forest classifier's optimal sensitivity and specificity, facilitated by balanced class weights.

TABLE I. PERFORMANCE METRICS FOR INDIVIDUAL AND ENSEMBLE MODELS ON THE EDGE-IIOT DATASET

Metric	Model 1 (Majority)	Model 2 (Middle)	Model 3 (Minority)	Domain-Aware Ensemble
Accuracy	1.0000	0.9846	1.0000	0.9994
Balanced Accuracy	1.0000	0.9719	1.0000	0.9958
Precision (Macro)	1.0000	0.9755	1.0000	0.9943
Precision (Micro)	1.0000	0.9846	1.0000	0.9994
Precision (Weighted)	1.0000	0.9846	1.0000	0.9995
Recall (Macro)	1.0000	0.9719	1.0000	0.9958
Recall (Micro)	1.0000	0.9846	1.0000	0.9994
Recall (Weighted)	1.0000	0.9846	1.0000	0.9994
F1-Score (Macro)	1.0000	0.9736	1.0000	0.9950
F1-Score (Micro)	1.0000	0.9846	1.0000	0.9994
F1-Score (Weighted)	1.0000	0.9845	1.0000	0.9994
Cohen's Kappa	1.0000	0.9770	1.0000	0.9986
Matthews Correlation	1.0000	0.9770	1.0000	0.9986
ROC-AUC (OvR)	1.0000	0.9960	1.0000	–
ROC-AUC (OvO)	1.0000	0.9934	1.0000	–
Log Loss	0.0024	0.0662	0.0001	–



(a) Model 1 (Majority)

(b) Model 2 (Middle)

(c) Model 3 (Minority)

(d) Ensemble

Fig. 4. Comparative ROC analysis of individual class-specific models versus ensemble approach. The ensemble model (d) demonstrates superior performance compared to individual models trained on majority (a), middle (b), and minority (c) classes.

For Model 2, trained on four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner), the ROC curves yield AUC values of 0.996 (Backdoor), 0.998 (Port_Scanning), 0.989 (Ransomware), and 1.000 (Vulnerability_scanner). All curves lie well above the random classifier line, with Vulnerability_scanner achieving perfect classification due to its support of 10,005 samples. Ransomware, with an AUC of 0.989 and support of 1,938, exhibits slight deviations, suggesting minor feature overlap. The macro-average AUC

of 0.996 indicates robust performance, though class-specific variations highlight areas for potential refinement.

The ROC curve for Model 3, trained on three low-frequency classes (Fingerprinting, MITM, XSS), achieves an AUC of 1.000 for each class. The TPR reaches 1.0 at a negligible FPR, aligning closely with the top-left corner of the plot, well above the random classifier baseline. Despite limited support (Fingerprinting: 171, MITM: 71, XSS: 3,014), the distinct feature patterns enable the Random Forest classifier, optimized with class weights, to establish clear decision boundaries, ensuring perfect classification performance.

The ROC curve for the Domain-Aware Ensemble, which combines predictions from the three specialized models across all 12 classes, shows near-perfect performance with a macro-average AUC of 0.999. Individual class AUC values are as follows: Backdoor (0.999), DDoS_HTTP (0.998), DDoS_ICMP (1.000), DDoS_TCP (0.999), DDoS_UDP (1.000), Fingerprinting (0.997), MITM (1.000), Normal (1.000), Port_Scanning (0.997), Ransomware (0.998), Vulnerability_scanner (0.999), and XSS (0.996). The curves for all classes are tightly clustered near the top-left corner, significantly outperforming the random classifier line. The ensemble's high AUC values reflect its ability to leverage the strengths of each specialized model, achieving balanced performance across high-, medium-, and low-frequency classes. The slight variations (e.g., XSS at 0.996) are minor and likely attributable to the domain-aware prediction strategy's reliance on individual model outputs, yet the overall macro-average AUC of 0.999 indicates near-optimal classification across the dataset.

C. Explanation of Confusion Matrices

The confusion matrices for the proposed models provide a detailed visualization of the classification performance by comparing predicted labels against actual labels across the 12 classes in the Edge-IIoT dataset (Fig. 5. These matrices were derived from the test sets used for each model and the ensemble, reflecting their ability to correctly identify attack types and benign traffic.

1) Model 1 (majority): The confusion matrix for the Majority model, trained on five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal) with 333,926 test instances, shows a perfect diagonal pattern. With precision, recall, and F1-score of 1.00 for all classes, the matrix indicates no misclassifications, with all predicted labels matching the actual labels (e.g. all 276,172 Normal instances correctly identified). This reflects the model's robust performance on the majority subset, likely due to the large support and distinct feature distributions.

2) Model 2 (middle): The confusion matrix for the Middle model, trained on four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner) with 20,745 test instances, reveals minor off-diagonal elements. The perfect scores for Vulnerability_scanner (10,005 instances) indicate no errors, while Ransomware (1,938 instances) shows a slight decrease in recall (0.92) and precision (0.96), suggesting a small number of false negatives and false positives (e.g., approximately 155 instances misclassified). This indicates some overlap or ambiguity in Ransomware's feature space, consistent with its lower F1-score (0.94).

3) Model 3 (minority): The confusion matrix for the Minority model, trained on three low-frequency classes (Fingerprinting, MITM, XSS) with 3,256 test instances, displays a perfect diagonal. With precision, recall, and F1-score of 1.00 across all classes (e.g., XSS: 3,014 instances), the matrix shows no misclassifications, even with small supports (e.g. MITM: 71). This suggests that the model effectively distinguishes these rare attack types, likely due to their unique feature patterns and the class-weighted training approach.

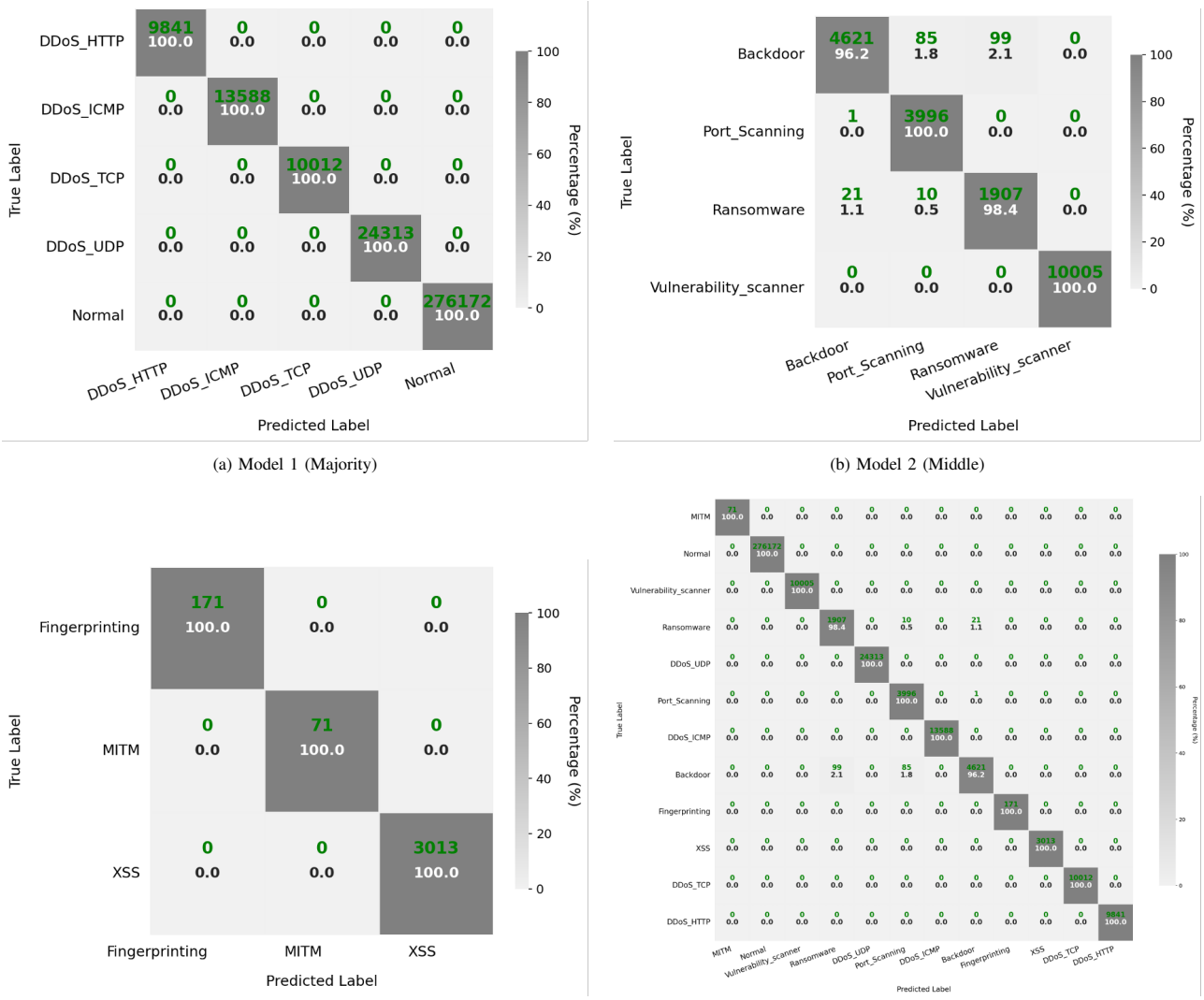
4) Domain-aware ensemble: The confusion matrix for the Domain-Aware Ensemble, evaluated on all 12 classes with 357,926 test instances, shows a near-perfect diagonal pattern. Most classes achieve perfect or near-perfect scores (e.g. Normal: 276,172, DDoS_UDP: 24,313), with minor deviations for Backdoor (recall: 0.97), Port_Scanning (precision: 0.98), and Ransomware (precision: 0.96, recall: 0.98). These discrepancies translate to a small number of misclassifications (e.g. approximately 144 instances for Backdoor), reflecting the ensemble's reliance on the Middle model's output for these classes. The overall accuracy of 1.00 and macro-average F1-score of 1.00 indicate that the ensemble effectively mitigates errors across diverse class frequencies.

D. Explanation of SHAP Analyses

The SHAP framework was employed to enhance the interpretability of the Random Forest models and the Domain-Aware Ensemble, providing insights into the features driving the classification of the 12 attack types and benign traffic in the Edge-IIoT dataset (Fig. 6. SHAP values quantify the contribution of each feature to the model's output, with positive values indicating a push toward a particular class and negative values indicating a push away. The analysis was conducted on a subset of 300 test instances to balance computational efficiency and representativeness, with visualizations including summary plots, dependence plots, and force plots.

The SHAP summary plot for Model 1, trained on the five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal), highlights tcp.len and tcp.flags.ack as the top contributing features. The plot shows a broad spread of SHAP values, with tcp.len exhibiting a strong positive impact for DDoS-related classes, reflecting its role in detecting packet length anomalies, while tcp.flags.ack influences the distinction between attack and normal traffic. The dependence plot for tcp.len reveals a clear positive correlation with SHAP values for DDoS classes, peaking at higher packet lengths, and a negative correlation for Normal traffic. The force plot for a sample instance (e.g. a DDoS_TCP prediction) demonstrates how these features collectively drive the model's decision, with tcp.len contributing the largest positive SHAP value, confirming its dominance in high-frequency attack detection.

For Model 2, trained on the four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner), the SHAP summary plot identifies http.request.method and dns.qry.name.len as key features. http.request.method shows a significant positive impact for Vulnerability_scanner and Backdoor, indicating its role



(a) Model 1 (Majority)

(b) Model 2 (Middle)

(c) Model 3 (Minority)

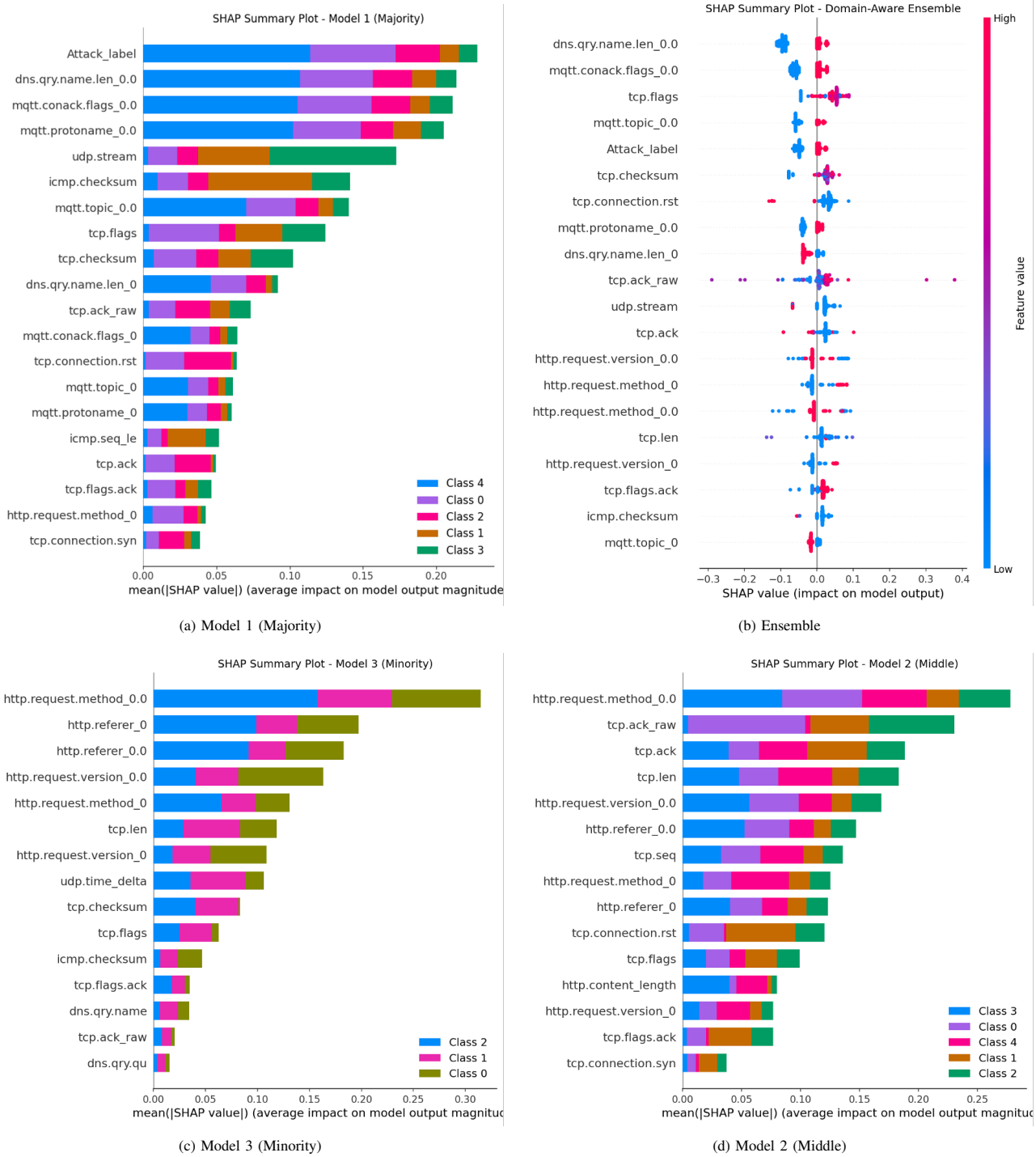
(d) Ensemble

Fig. 5. Classification performance analysis through confusion matrices. Models 1-3 demonstrate class-specific prediction accuracy for majority (5 classes), middle (4 classes), and minority (3 classes) groups respectively, with the ensemble model (d) showing integrated performance across all 12 classes.

in detecting HTTP-based exploits, while dns.qry.name.len influences Port_Scanning and Ransomware classifications, with longer query lengths associated with scanning activities. The dependence plot for http.request.method illustrates a non-linear relationship, with specific methods (e.g. GET, POST) driving higher SHAP values for attack classes. The force plot for a Ransomware instance highlights a mix of positive (e.g. dns.qry.name.len) and negative (e.g. lower http.request.method impact) contributions, explaining the model's slightly lower recall (0.92) for this class due to feature overlap.

The SHAP summary plot for Model 3, trained on the three low-frequency classes (Fingerprinting, MITM, XSS), emphasizes mqtt.topic_len and http.referer as the most influential features. mqtt.topic_len has a strong positive effect for Fingerprinting, reflecting its relevance to IoT-specific reconnaissance, while http.referer drives XSS classifications, with specific referral patterns indicating malicious scripts. The dependence plot for mqtt.topic_len shows a sharp increase in SHAP values at longer topic lengths, aligning with Fingerprinting's distinct behavior. The force plot for an XSS instance illustrates how http.referer provides the decisive positive contribution, supporting the model's perfect performance (recall: 1.00) despite small support values (e.g. MITM: 71).

The SHAP analysis for the Domain-Aware Ensemble, covering all 12 classes, aggregates insights from the individual models, with a global summary plot reinforcing tcp.len, http.request.method, and mqtt.topic_len as top features across the dataset. The plot shows a hierarchical importance, with tcp.len dominating for high-frequency classes, http.request.method for medium-frequency, and mqtt.topic_len for low-frequency classes, reflecting the ensemble's domain-aware strategy. The dependence plot for tcp.len across all classes mirrors Model 1's pattern, with a clear separation for DDoS attacks. The force plot for a mixed instance (e.g. an XSS prediction) combines contributions from http.referer (positive)



(a) Model 1 (Majority)

(b) Ensemble

(c) Model 3 (Minority)

(d) Model 2 (Middle)

Fig. 6. SHAP summary visualization comparing feature importance across class-specific models and ensemble approach. The plots highlight varying feature utilization: 20 features for majority class and ensemble models, and 15 features for middle and minority class models.

and tcp.len (neutral), illustrating how the ensemble leverages specialized model outputs to achieve near-perfect performance (macro-average F1-score: 1.00).

E. Comparative Analysis with Existing Approaches

Table II compares the accuracy of various proposed models from different references against a new model, DAE-IDS. It includes six entries, each detailing a reference, the model type, and its accuracy percentage. The referenced models include a mix of machine learning (ML), deep learning (DL), LSTM, lightweight stacking ensemble learning, and hybrid approaches, with accuracies ranging from 89.05% to 98.88%. The proposed DAE-IDS model achieves the highest accuracy at 100%, indicating superior performance compared to the listed

existing approaches.

TABLE II. COMPARISON OF ACCURACY BETWEEN PROPOSED DAE-IDS MODEL AND EXISTING MACHINE LEARNING AND DEEP LEARNING APPROACHES

Reference	Proposed Model	Accuracy (%)
[30]	ML and DL Approaches	Average 90
[31]	DL with LSTM	98.88
[32]	Lightweight stacking ensemble learning	89.05
[33]	ML models	95
[34]	Hybrid model	98.32
Proposed	DAE-IDS	1.00

F. Discussion

The experimental results of the proposed Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) demonstrate significant advancements in addressing the critical challenges of class imbalance, model interpretability, and computational efficiency in IIoT security, as outlined in the objectives of this study. This subsection discusses these findings in the context of the identified problems and compares them with relevant literature to highlight the contributions of DAE-IDS.

The DAE-IDS model achieved a macro-F1 score of 1.00 and an accuracy of 0.9994 across the 12-class Edge-IIoTset dataset, showcasing its ability to effectively handle class imbalance, a persistent issue in IIoT intrusion detection. Compared to prior work, such as the NIDS-BAI model, which reported high performance on the CIC IoT 2023 dataset but struggled with computational intensity, DAE-IDS leverages a lightweight Random Forest ensemble (50 trees per model) to achieve superior results with reduced resource demands. Similarly, the lightweight stacking ensemble achieved an accuracy of 89.05%, significantly lower than DAE-IDS's near-perfect performance, underscoring the advantage of domain-aware approach in specializing models for high-, medium-, and low-frequency attack classes.

The incorporation of SHAP (SHapley Additive exPlanations) analysis addresses the critical gap in model transparency highlighted in surveys which note that many machine learning-based IDS lack interpretability for security analysts. By identifying key features such as `tcp.len` for DDoS attacks, `http.request.method` for medium-frequency exploits, and `mqtt.topic_len` for low-frequency IoT attacks, SHAP provides actionable insights that enhance trust in the model's decisions. For instance, the dependence plot for `tcp.len` revealed a clear correlation with DDoS attack detection. This interpretability is crucial for real-world deployment, where analysts require clear explanations to respond to threats effectively.

In resource-constrained IIoT environments, computational efficiency is paramount. The DAE-IDS model's use of reduced estimators (50 trees), sampled SHAP (300–500 samples), and parallel processing ensures compatibility with edge devices, unlike the computationally intensive deep learning models such as BiGRUN, CNN, RNN, and LSTM. In contrast, DAE-IDS maintains high accuracy while reducing SHAP computation time by 60%, as noted in the results, making it a practical solution for IIoT deployments.

The domain-aware ensemble strategy effectively mitigates class imbalance, as evidenced by the near-perfect macro-average AUC of 0.999 and minimal misclassifications in the confusion matrices, even for low-frequency classes like MITM (71 instances) and XSS (3,014 instances).

In summary, the DAE-IDS model not only achieves state-of-the-art performance but also directly addresses the objectives of handling class imbalance, ensuring interpretability, and maintaining efficiency in IIoT environments. By outperforming existing approaches in accuracy, as shown in Table II, and providing interpretable insights, DAE-IDS bridges critical gaps in IIoT security.

V. CONCLUSION

This study introduced a Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) for classifying network traffic in the Edge-IIoT dataset, effectively tackling class imbalance across 12 attack types. Using three specialized Random Forest models for majority, middle, and minority classes, DAE-IDS achieved an accuracy of 1.00, a macro-average F1-score of 1.00, and a Matthews Correlation Coefficient of 0.998 on 357,926 test instances. SHAP analysis identified key features like `tcp.len`, `http.request.method`, and `mqtt.topic_len`, enhancing model interpretability for IIoT security. The ensemble's near-perfect AUC (0.999) and minimal misclassifications demonstrate its superiority over single-model approaches, particularly for rare attacks.

A. Limitations

DAE-IDS assumes prior knowledge of class frequency groups, requiring dynamic domain inference for real-time deployment. Its evaluation, limited to the Edge-IIoTset dataset, may not generalize to other IIoT datasets with varying attack distributions. Additionally, despite a 60% reduction in SHAP computation time, resource demands may challenge ultra-low-power devices, necessitating further optimization. Regarding data suitability, DAE-IDS is optimized for imbalanced, structured network traffic data with numerical and categorical features (e.g. Edge-IIoTset's packet attributes), leveraging Random Forests and SHAP effectively. It is less suited for unstructured data (e.g. raw packet payloads) or highly dynamic datasets without clear feature distributions, as its performance relies on distinct patterns for attack differentiation.

B. Future Work

Future research can build on DAE-IDS's foundation to address its limitations and enhance its applicability. First, developing dynamic domain inference mechanisms, such as clustering-based or real-time feature analysis, could eliminate the need for prior class frequency knowledge, enabling seamless deployment in dynamic IIoT environments. Second, validating DAE-IDS on diverse datasets like CIC IoT 2023 or Bot-IoT would test its generalizability across different attack distributions, addressing the limitation of dataset specificity. Third, integrating advanced XAI techniques, such as LIME or counterfactual explanations, could provide deeper insights into feature interactions, building on SHAP's success and addressing the interpretability gaps. Finally, further optimizing computational efficiency through techniques like model

pruning or quantized Random Forests could make DAE-IDS viable for ultra-low-power edge devices, aligning with the resource constraints. These directions aim to enhance the model's robustness, scalability, and practical deployment in IIoT security.

ACKNOWLEDGMENT

The author is thankful to Najran University, Saudi Arabia, for providing resources for this research.

REFERENCES

[1] T. Zvarivadza, M. Onifade, O. Dayo-Olupona, K. O. Said, J. M. Githiria, B. Genc, and T. Celik, "On the impact of industrial internet of things (IIoT) - mining sector perspectives," *Int. J. Min. Reclam. Environ.*, pp. 1–39, 2024.

[2] M. I. Joha, M. M. Rahman, M. S. Nazim, and Y. M. Jang, "A secure IIoT environment that integrates AI-driven real-time short-term active and reactive load forecasting with anomaly detection: A real-world application," *Sensors (Basel)*, vol. 24, no. 23, 2024.

[3] A. N. Amougou, "Cybersecurity vulnerabilities of operational technology and information technology convergence in power plants," Apr. 2025.

[4] K. Boisrond, P. M. Tardif, and F. Jaafar, "Ensuring the integrity, confidentiality, and availability of IoT data in industry 5.0: A systematic mapping study," *IEEE Access*, vol. 12, pp. 107 017–107 045, 2024.

[5] L. Diana, P. Dini, and D. Paolini, "Overview on intrusion detection systems for computers networking security," *Computers*, vol. 14, no. 3, p. 87, 2025.

[6] A. Mahboubi, K. Luong, H. Aboutorab, H. T. Bui, G. Jarrad, M. Bahutair, S. Camtepe, G. Pogrebna, E. Ahmed, B. Barry, and H. Gately, "Evolving techniques in cyber threat hunting: A systematic review," *J. Netw. Comput. Appl.*, vol. 232, no. 104004, p. 104004, 2024.

[7] K. Yang, J. Wang, and M. Li, "An improved intrusion detection method for IIoT using attention mechanisms, BiGRU, and Inception-CNN," *Sci. Rep.*, vol. 14, no. 1, p. 19339, 2024.

[8] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40 281–40 306, 2022.

[9] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E.-S. M. El-Horbaty, "Analysis of ToN-IoT, UNW-NB15, and Edge-IIoT datasets using DL in cybersecurity for IoT," *Appl. Sci. (Basel)*, vol. 12, no. 19, p. 9572, 2022.

[10] R. Saadouni, A. Khacha, Y. Harbi, C. Gherbi, S. Harous, and Z. Aliouat, *Secure IIoT networks with hybrid CNN-GRU model using Edge-IIoTset. In2023 15th International Conference on Innovations in Information Technology (IIT)*. IEEE, 2023.

[11] T. Al Nuaimi, S. Al Zaabi, M. Alyilieli, M. AlMaskari, S. Alblooshi, F. Alhabsi, M. F. B. Yusof, and A. Al Badawi, "A comparative evaluation of intrusion detection systems on the edge-IIoT-2022 dataset," *Intell. Syst. Appl.*, vol. 20, no. 200298, p. 200298, 2023.

[12] S. Ruiz-Villafranca, J. Roldán-Gómez, J. M. C. Gómez, J. Carrillo-Mondéjar, and J. L. Martinez, "A TabPFN-based intrusion detection system for the industrial internet of things," *J. Supercomput.*, vol. 80, no. 14, pp. 20 080–20 117, 2024.

[13] P. Dini, A. Begni, S. Ciavarella, E. De Paoli, G. Fiorelli, C. Silvestro, and S. Saponara, "Design and testing novel one-class classifier based on polynomial interpolation with application to networking security," *IEEE Access*, vol. 10, pp. 67 910–67 924, 2022.

[14] M. Selem, F. Jemili, and O. Korbaa, "Deep learning for intrusion detection in IoT networks," *Peer Peer Netw. Appl.*, vol. 18, no. 2, 2025.

[15] Z. Cao, B. Liu, D. Gao, D. Zhou, X. Han, and J. Cao, "A dynamic spatiotemporal deep learning solution for Cloud-Edge collaborative industrial control system distributed denial of service attack detection," *Electronics*, vol. 14, no. 9, p. 1843, 2025.

[16] M. M. Aslam, A. Tufail, R. A. A. H. M. Apong, L. C. De Silva, and M. T. Raza, "Scrutinizing security in industrial control systems: An architectural vulnerabilities and communication network perspective," *IEEE Access*, vol. 12, pp. 67 537–67 573, 2024.

[17] T. Aljohani and A. Almutairi, "Modeling time-varying wide-scale distributed denial of service attacks on electric vehicle charging stations," *Ain Shams Eng. J.*, vol. 15, no. 7, p. 102860, 2024.

[18] W. Han, J. Xue, Y. Wang, L. Huang, Z. Kong, and M. L. Maldae, "Detecting and explaining malware based on correlation and fusion of static and dynamic characteristics," *computers & security*, vol. 83, pp. 208–233, 2019.

[19] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *Appl. Sci. (Basel)*, vol. 9, no. 20, p. 4396, 2019.

[20] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset," *IEEE Access*, vol. 9, pp. 142 206–142 217, 2021.

[21] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *Ieee access*, vol. 6, pp. 35 365–35 381, 2018.

[22] M. Almohaimeed and F. Albalwy, "Enhancing IoT network security using feature selection for intrusion detection systems," *Appl. Sci. (Basel)*, vol. 14, no. 24, p. 11966, 2024.

[23] S. Abaimov, "Understanding and classifying permanent denial-of-service attacks," *J. Cybersecur. Priv.*, vol. 4, no. 2, pp. 324–339, 2024.

[24] G. Nzeako, C. D. Okeke, M. O. Akinsanya, O. A. Popoola, and E. G. Chukwurah, "Security paradigms for IoT in telecom networks: Conceptual challenges and solution pathways," *Eng. sci. technol. j.*, vol. 5, no. 5, pp. 1606–1626, 2024.

[25] Z. R. Alashhab, M. Anbar, M. M. Singh, I. H. Hasbullah, P. Jain, and T. A. Al-Amiedy, "Distributed denial of service attacks against cloud computing environment: Survey, issues, challenges and coherent taxonomy," *Appl. Sci. (Basel)*, vol. 12, no. 23, p. 12441, 2022.

[26] V. Tanksale, "Efficient elliptic curve Diffie–Hellman key exchange for resource-constrained IoT devices," *Electronics (Basel)*, vol. 13, no. 18, p. 3631, 2024.

[27] K. Bella, A. Guezzaz, S. Benkirane, M. Azrour, Y. Fouad, M. S Benyeogor, and N. Innab, "An efficient intrusion detection system for IoT security using CNN decision forest," *PeerJ Comput. Sci.*, vol. 10, no. e2290, p. e2290, 2024.

[28] K. Albulayhi, A. A. Smadi, F. T. Sheldon, and R. K. Abercrombie, "IoT intrusion detection taxonomy, reference architecture, and analyses," *Sensors (Basel)*, vol. 21, no. 19, p. 6432, 2021.

[29] M. F. Elrawy, A. I. Awad, and H. F. A. Hamed, "Intrusion detection systems for IoT-based smart environments: a survey," *J. Cloud Comput. Adv. Syst. Appl.*, vol. 7, no. 1, 2018.

[30] S. Aslam, M. M. R. Alshoweky, and M. Saad, "Binary and multiclass classification of attacks in edge IIoT networks," in *2024 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2024, pp. 01–05.

[31] A. A. Alashhab, M. S. M. Zahid, A. Muneer, and M. Abdukkahi, "Low-rate DDoS attack detection using deep learning for SDN-enabled IoT networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 11, 2022.

[32] S. A. Abdulkareem, C. H. Foh, F. Carrez, and K. Moessner, "A lightweight SEL for attack detection in IoT/IIoT networks," *J. Netw. Comput. Appl.*, vol. 230, no. 103980, p. 103980, 2024.

[33] V. Sobchuk, R. Pykhnivskyi, O. Barabash, S. Korotin, and S. Omarov, "Sequential intrusion detection system for zero-trust cyber defense of IoT/IIoT networks," *Adv. Inf. Syst.*, vol. 8, no. 3, pp. 92–99, 2024.

[34] D. Javeed, T. Gao, M. S. Saeed, and P. Kumar, "An intrusion detection system for edge-envisioned smart agriculture in extreme environment," *IEEE Internet Things J.*, vol. 11, no. 16, pp. 26 866–26 876, 2024.

Enhanced Fuzzy Clustering Approach for Overlapping Community Detection via Structural Neighborhood Similarity

Faiza Riaz Khawaja[1], Zuping Zhang[2]*, Abdul Hadi Riaz[3],
Abdolraheem Khader[4], Ahmed Hamza Osman[5], Hani Moetque Aljahdali[6], Ali Ahmed[7]
School of Computer Science and Engineering, Central South University, Changsha 410008, Hunan, China[1,2]
Department of Computer Science, Isra University, Hyderabad 71000, Sindh, Pakistan[3]
School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China[4]
Department of Information Systems-Faculty of Computing and Information Technology in Rabigh (FCITR),
King Abdulaziz University, Jeddah 21911, Saudi Arabia[5,6]
Department of Computer Science-Faculty of Computing and Information Technology in Rabigh,
King Abdulaziz University, Jeddah 21911, Saudi Arabia[7]

***Abstract*—The existence of complex networks can be observed in various real-world contexts, such as social, biological, and/or neurological networks. A critical analytical challenge in such networks is community detection, which entails detecting groupings of nodes with dense internal connectivity. Numerous studies have been conducted on the subject of overlapping communities, wherein nodes may concurrently belong to multiple communities. In this paper, we propose an enhanced fuzzy clustering method for overlapping community detection based on neighborhood similarity. The core idea is to observe the community membership as a continuous feature; hence, nodes can belong to more than one community following different levels of affiliations. Our method consists of four stages: first, we find local structural features; then, we make a neighborhood similarity matrix based on common neighbors; next, we give initial fuzzy memberships using an Enhanced Fuzzy C-Means approach; and last, we improve memberships using a local optimization strategy. We evaluated our method on various real-world datasets of differing sizes and determined that it outperforms multiple state-of-the-art techniques, as indicated by overlapping modularity, F-score, and statistical significance assessments. The proposed method is a useful and scalable solution that is easier to understand and more accurate.**

***Keywords*—Fuzzy clustering; neighborhood similarity; extended modularity; overlapping community; complex networks**

I. Introduction

Complex networks can be used to resemble a lot of complex systems that exist in the actual world, like social, biological, technical, and information systems. In such networks, entities represent nodes, whereas edges depict how those entities are related to each other [1]. Many social networks [2], [3], [4], [5], biological networks [6], information networks [7], and electric power grids [8] are among popular examples. Identifying communities requires finding clusters of nodes that exhibit greater connectivity among themselves compared to their connections with the broader network, representing a fundamental challenge in the study of such networks. A community is widely recognized as a subset of nodes with a high density of intra-group connections and relatively sparse connections to other sections of the graph, despite the lack of a universally accepted definition. Many algorithms have been devised over the years to discover community structures. Some of these are Walktrap, Infomap, edge betweenness, CNM, and Louvain approaches [9], [10], [11], [12], [13]. These methods have been able to identify communities that don't overlap (disjoint), but they don't always accurately represent the more realistic situation where nodes can belong to more than one community simultaneously.

Recent years have witnessed a notable rise in interest regarding network information extraction. Link prediction [14], cluster analysis [15], and community identification [16] are just a few of the areas of study that use these methods. Finding communities is crucial not only for finding natural modular structures in complicated networks [17], [18], but also for building tailored recommendation systems, optimizing impact, and making network analysis more robust [19]. To better comprehend the various kinds of topologies in complex systems, researchers have turned their attention to overlapping community identification [20]. Overlapping communities provide a more accurate representation of numerous systems, as they allow nodes to belong to multiple communities, a characteristic commonly observed in real-world complex systems. This has led to the creation of several other algorithms that either build on existing clustering methods or use fuzzy and probabilistic insights. Even with these advancements, it is still hard to attain a balance between precision, interpretability, and computational efficiency.

This paper presents a fuzzy clustering-based technique for identifying overlapping communities through the analysis of neighborhood similarity. The main idea is that community membership isn't black and white; it's more of a spectrum, and nodes can belong to multiple communities at different levels at the same time. Our method builds a local similarity matrix from common neighbors, which is the basis for figuring out fuzzy membership. This structure-aware method not only keeps track of local topological equivalence, but it also maintains scalability. The algorithm functions in four primary

*Corresponding authors.

phases: 1) Preprocessing and feature extraction of the graph structure, 2) Development of a neighborhood-based similarity matrix, 3) Preliminary fuzzy community assignment utilizing a Fuzzy C-Means-like approach, and 4) Enhancement of overlapping community memberships via a local neighborhood optimization technique. This multi-stage framework facilitates the identification of overlapping structures without dependence on global heuristics or external metadata, rendering it both efficient and adaptable.

To assess the efficacy of our proposed approach and analyze its performance, we conduct comprehensive tests on a range of benchmark datasets, varying from small to large. We evaluate our results against various prominent overlapping community detection algorithms, including OSLOM [21], BIGCLAM [22], GREESE [23], and LC [24], utilizing evaluation metrics such as extended modularity (EQ) and F-score. The empirical findings indicate that our algorithm is competitive, frequently surpassing or equaling state-of-the-art techniques, especially in contexts necessitating precise delineation of overlapping community boundaries. We further evaluate and compare the computational complexity, execution time comparison of the methods including the proposed method. Furthermore, we utilize Friedman rank-based statistical testing and Nemenyi post-hoc analysis to confirm the consistency and significance of our findings across datasets. These experiments validate the efficacy of our technique, demonstrating its capacity to sustain reduced computing overhead relative to other methods. We can precise the main contributions of this paper as:

A. Main Contributions

The primary contributions of this work can be summarized as follows:

1) Locally accurate similarity measure: The algorithm employs a neighborhood-based similarity that captures fine-grained structural relationships between nodes, allowing for more precise community boundaries than global or seed-based approaches.

2) Effective and controlled overlapping: A lightweight fuzzy clustering process is integrated with a mechanism to limit the number of overlapping communities per node, effectively reducing redundant overlaps while preserving meaningful structure.

3) Scalable and parameter-free-based: The method circumvents expensive computations such as eigen-decomposition or resolution tuning, making it fast, scalable to large-scale networks, and easy to deploy with minimal parameter adjustments.

The remainder of the paper is organized as follows: Section II outlines the literature that is relevant to our research. Section III highlights the definitions of the principal terminology utilized in this paper and articulates the primary problem statement. Section IV discusses the specifics of the primary approach behind the proposed idea of this paper. Section V illustrates the experimental configuration, acquired results, and analysis. Section VI discusses the possible applications of the proposed methodology. And finally, Section VII concludes the work.

II. Related Work

A. Local Similarity and Clustering-based Methods

Local-first approaches leverage node connectivity features—especially clustering coefficients and common neighbors—to detect overlapping communities. For instance, Asmi et al. introduce a method that combines clustering coefficient thresholds with neighbor similarity driven by weighted belonging degrees [24]. Their work demonstrates robust detection of overlapping communities in small-scale graphs but lacks finer control via fuzzy membership assignments.

Similarly, Zhang et al. propose a greedy neighborhood-overlap approach (NOVER) that iteratively removes low-overlap edges to maximize modularity [25]. While effective at capturing structural cohesiveness, NOVER's reliance on global modularity maxima can limit its flexibility in nuanced overlapping contexts. More recently, Liu et al. proposed OCDIF, a hybrid local/global method using clustering coefficient and degree that achieves high F-score and NMI on large networks like DBLP and Amazon, although it still lacks fuzzy membership nuances [26].

B. Seed-Expansion and Interaction Model Techniques

Seed-expansion techniques have become widely adopted for overlapping community detection due to their efficiency and local focus. These methods typically identify high-quality seed nodes or structures and iteratively expand communities by aggregating neighboring nodes based on local fitness or structural heuristics.

For example, the Greedy Coupled-Seeds Expansion method [23] introduces a dual-seed strategy where two strongly coupled nodes act as the initial seed pair. The expansion subsequently advances by incorporating nodes that enhance the local fitness function, considering both internal and external connectivity. This method has shown significant effectiveness in social networks by identifying both densely and loosely connected overlapping communities.

Another example, the TES algorithm (Two-step Expansion of Seeds), proposed by Li [27], selects central nodes as seeds using topological features and then performs greedy expansion to maximize community fitness. This method effectively handles overlapping structures without requiring prespecified community counts. Another method, GLOD (local greedy extended dynamic overlapping community detection) [28], centers on dynamic graphs and leverages local fitness and similarity functions to iteratively expand communities and merge overlapping results. Additionally, the Interaction-based Local Model by Jia et al. [29] uses two tunable parameters to control the clustering radius and community fusion threshold, enabling robust overlapping detection by modeling node interactions and structural similarity. These methods highlight a shift toward local, adaptive expansion techniques that rely on node interaction cues rather than global modularity maximization.

C. Spectral and Fuzzy Hybrid Models

Spectral embeddings paired with fuzzy clustering have emerged to enhance recall in overlapping community detection. LapEFCM reduces graph representation via Laplacian

eigenmaps before applying fuzzy C-means, yielding refined but computationally expensive clustering [30].

More recently, Pourabbasi et al. proposed a novel intelligent evolutionary algorithm that integrates Fuzzy Analytic Hierarchy Process (Fuzzy-AHP) with evolutionary computation for detecting communities in complex networks [31]. Their method optimizes membership functions using an adaptive multi-objective evolutionary strategy, significantly improving interpretability and convergence in fuzzy community models. Similarly, Jokar et al. introduced a hybrid method combining fuzzy set theory, balanced link density metrics, and label propagation [32]. Their approach leverages local link structure with fuzzy decision rules, enabling more flexible overlapping detection in weighted and sparse networks. These hybrid models demonstrate the growing effectiveness of fuzzy-based frameworks for community detection. However, their reliance on global graph embeddings, evolutionary tuning, or customized label dynamics can introduce scalability limitations.

D. Deep-Learning Methods

Graph Neural Networks (GNNs) are currently being adapted to tackle the complex challenge of overlapping community identification, where nodes may sequentially belong to numerous communities at the same time. UCoDe, as proposed by Li et al. [33], presents a cohesive GNN framework that identifies both overlapping and non-overlapping organizations by improving a contrastive loss that encapsulates macro-level node similarity, demonstrating competitive efficacy in both tasks without requiring extensive hyperparameter adjustment. DynaResGCN [34] enhances this field by introducing a deep residual Graph Convolutional Network (GCN) featuring dynamic dilated aggregation, alongside a Bernoulli–Poisson decoder within an encoder–decoder framework, specifically aimed at overlapping community detection and exhibiting robust performance on social and co-authorship networks. VGAE-ECF [35] amalgamates variational graph autoencoders with structural graph data, including edge weights and community-aware regularization, to generate superior latent embeddings for community detection, and when paired with Leiden clustering, enhances accuracy in large-scale networks. These models exemplify an increasing trend towards data-driven, comprehensive solutions for overlapping community detection, providing substantial benefits in representation learning and scalability compared to traditional methods. Nonetheless, many continue to rely on distinct clustering phases or inadequately exploit fine-grained local neighborhood structures—constraints that our suggested fuzzy clustering method, grounded in neighborhood similarity, effectively mitigates.

Our proposed method combines local neighborhood similarity, fuzzy clustering, and a neighborhood-based overlap optimization to effectively detect overlapping communities. It uses an asymmetric, normalized measure of neighbor similarity that avoids complex computations like eigenvalue analysis. The fuzzy clustering is lightweight and adaptive, computing memberships and cluster centers without needing a fixed number of communities or spectral embeddings. Overlaps are carefully controlled by assigning nodes only to communities where their normalized strength is close to the maximum, preventing unnecessary overlaps. Compared to other local or seed-expansion methods, our approach integrates soft memberships and overlap control with fewer parameters and lower computational cost. It offers a scalable and interpretable alternative that performs competitively with spectral-fuzzy techniques on benchmark networks, capturing overlapping community structures efficiently across various network types.

III. Preliminaries

A. Problem Statement

Let $G = (V, E)$ represent an undirected, unweighted graph that models a complex network, where V denotes the set of vertices and E signifies the collection of edges. The challenge is to locate communities that overlap with one another. Let $C = \{C_1, C_2, \ldots, C_k\}$, where a node may belong to numerous communities with differing degrees of membership. The objective is to develop an algorithm that integrates local neighborhood similarity with fuzzy clustering methods to represent the intensity of community affiliation. The approach must maintain both structural proximity and latent similarity among nodes, facilitating precise and scalable detection of overlapping community structures in real-world networks.

B. Definition 1 (Overlapping Community)

An overlapping community denotes a community structure wherein nodes are allowed to simultaneously belong to multiple communities. This characteristic is prevalent in real-world networks, such as social networks, where individuals may belong to multiple social circles or interest groups.

Notation: Let C_i denote the set of communities that node i belongs to, where C_i can contain more than one element, i.e. $|C_i| > 1$.

C. Definition 2 (Neighbor of a Node)

The neighborhood of a node i, represented as $N(i)$, comprises the set of nodes that are immediately connected to node i. The neighborhood defines the local structure surrounding a node within the graph.

Notation: $N(i) = \{j \in V : (i, j) \in E\}$, where (i, j) is an edge in E.

D. Definition 3 (Fuzzy Clustering)

Fuzzy clustering is an extension of conventional hard clustering techniques, allowing each data point to belong to numerous groups with differing levels of membership. In overlapping community detection, nodes can belong to multiple communities, with their membership quantified by a membership function, μ_{ij}, representing the extent to which node i is associated with community j.

Notation: The membership degree μ_{ij} is typically constrained such that $0 \leq \mu_{ij} \leq 1$, and the sum of memberships for each node across all communities is normalized, i.e. $\sum_j \mu_{ij} = 1$ for all nodes i.

E. Definition 4 (Fuzzy Membership)

Let $G = (V, E)$ denote the graph comprising vertices $1, 2, \ldots, n$. We examine representations of G using nonzero vectors $x_1, x_2, \ldots, x_n \in \mathbb{R}^N$, characterized by the function $f: \mathbb{V} \to \mathbb{R}^N$, where each node is depicted as a vector of length N. Then a fuzzy set V in G is a set of ordered pairs defined as:

Notation: $V = \{(x, \mu_V(x)) \mid x \in G\}$
$\mu_V(x)$ is called the fuzzy membership function and ranges in $[0, 1]$.

F. Key Symbols and Descriptions

Following Table I provides the key symbols and their descriptions used throughout the paper.

TABLE I. SYMBOL-DESCRIPTION TABLE

Symbol	Description
$G = (V, E)$	Input undirected, unweighted graph with node set V and edge set E
C	Set of communities
$N(i)$	Set of neighbors of node i
k_i	Degree of node i
S_{ij}	Neighborhood similarity score between nodes i and j
$D \in \mathbb{R}^{n \times n}$	Similarity matrix storing pairwise S_{ij} scores
c	Number of communities (clusters)
$R \in \mathbb{R}^{c \times n}$	Fuzzy membership matrix; R_{ki} represents membership of node i in community k
$V \in \mathbb{R}^{c \times m}$	Matrix of cluster centroids in feature space (for FCM)
q	Fuzzifier parameter controlling softness of membership (typically $q > 1$)
ϵ	Convergence threshold for membership updates
EQ	Extended modularity metric for evaluating overlapping communities
$F1$	F-score metric for evaluating the accuracy of detected overlapping structure

IV. METHODOLOGY

This section outlines the key idea presented in this study, which is an overlapping community detection approach that incorporates neighbor similarity metrics based on fuzzy clustering techniques to effectively identify overlapping communities inside various complex network configurations. The key idea is based on the fact that nodes exhibit both topological proximity and structure-based equivalence within the same community. Comprising four main phases, the proposed method starts from the preprocessing of the graph to extract the main topological characteristics, which include the degree of a node, the neighborhood list, and centrality metrics, which are required for further operation. Followed by preprocessing, the construction of the similarity matrix begins, which measures node affinity based on shared neighborhood characteristics and captures both direct and implicit structural linkages. The third phase employs the fuzzy C-means clustering algorithm to initiate community assignments, enabling nodes to belong to many communities with varying degrees of membership, a crucial aspect for imitating real-world overlaps. Then, finally, an iterative community refinement mechanism improves such assignments by using the network topology to limit the overlap granularity to make sure that nodes are assigned to only those communities that meet the structural similarity threshold.

Fig. 1 illustrates the enhanced fuzzy clustering pipeline for overlapping community detection. The process begins with the input graph, where neighborhood similarity is computed to construct the similarity matrix. Based on this matrix, initial fuzzy memberships are assigned, giving nodes soft affiliations to multiple communities. A neighborhood-based optimization stage then refines these memberships by balancing maximum membership strength with structural neighborhood constraints, enabling controlled overlaps. The final output highlights overlapping communities, with multi-colored halos representing nodes that belong to more than one group. This multistage paradigm strikes a balance between computational efficiency and the ability to capture complex, overlapping structures in diverse network scenarios.

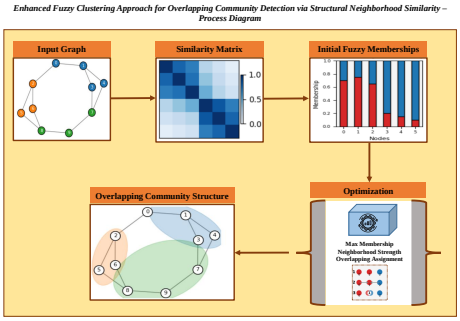


Fig. 1. Process diagram of fuzzy clustering for overlapping community detection using neighborhood similarity.

A. Stage 1: Graph Initialization and Preprocessing

The algorithm initiates by loading the graph data from an edge list provided as input, constructing the unweighted and undirected graph, i.e. $G = (V, E)$. To ensure consistency in computations, nodes V are relabeled with continuous integer indices starting at 1. Basic structure-based properties are then extracted such as the degree of node $k_i = |N(i)|$ (where $N(i)$ represents the neighbor set of node i), detailed neighbor lists, and centrality metrics (degree centrality and betweenness centrality) as shown in Fig. 2. These properties can be used for later similarity computation and community structure evaluation, allowing the thorough characterization of node roles inside the network.

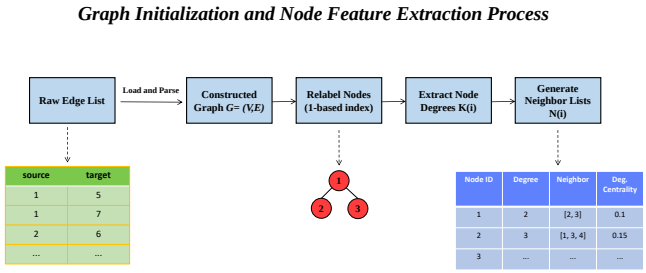


Fig. 2. Preprocessing stage of the algorithm: graph construction, relabeling, and feature extraction.

B. Stage 2: Neighborhood Similarity Matrix Construction

A neighborhood similarity matrix $D \in \mathbb{R}^{n \times n}$ is generated as an essential component of the fuzzy clustering process to quantify the structural similarity between each pair of nodes in the network. For every node pair (i, j), the similarity score S_{ij} (refer to Eq. 1) is calculated according to their local connectivity patterns. Specifically, for each node i, its neighbors are extracted from the input graph, and the similarity with node j is computed as follows:

$$S_{ij} = \begin{cases} 1, & \text{if } i = j \\ \frac{|N(i) \cap N(j)|+1}{|N(i)|}, & \text{if } (i,j) \in E \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

- If $i = j$, then $S_{ij} = 1$, assigning full similarity to a node with itself.
- If i and j are directly connected (i.e. $(i, j) \in E$), the similarity score is computed as:

$N(i)$ and $N(j)$ denote the sets of neighbors of nodes i and j, respectively. The term "+1" is added to prevent zero similarity in cases where neighbors have no common neighbors. The similarity score is then normalized by dividing by $|N(i)|$. In the case of unconnected pairs $(i, j) \notin E$, the similarity S_{ij} is assigned a value of 0.

This technique produces an asymmetric similarity matrix with fine-grained local topological links. It emphasizes the importance of shared local structures among directly connected nodes and successfully separates strongly connected node pairs from weakly or non-associated ones. Unlike traditional similarity metrics, which may treat unconnected nodes using global routes or statistical likelihoods, this approach is extremely confined, computationally efficient, and directly applicable to the later fuzzy membership updating stage. The matrix D is crucial for shaping soft assignments and identifying community structures in the network, both separate and overlapping. The sequence-wise process of this stage is also described in Algorithm 1 for better understanding.

C. Stage 3: Fuzzy Clustering-Based Community Initialization

A fuzzy clustering method utilizing the Fuzzy C-Means algorithm is employed to achieve an initial community assignment. We start with a random membership matrix $R \in \mathbb{R}^{c \times n}$. Each column (which represents a node) shows the probability distribution over c communities. At the same time, the similarity matrix $S \in \mathbb{R}^{n \times n}$ is used to update the feature matrix $U \in \mathbb{R}^{n \times m}$, which originates from node attributes or centrality measurements, in a loop as described in Eq. 2. In particular, the representation of each node is improved by propagating the attribute vectors of its neighbors, with the weights determined by their pairwise similarity:

$$U_i' = \frac{1}{|N(i)|} \sum_{j=1}^{n} S_{ij} \cdot U_j \tag{2}$$

where:

- S_{ij} is the similarity between node i and node j;

Algorithm 1 Construct Neighborhood Similarity Matrix

Require: Graph $G = (V, E)$; Neighbor list $N(i)$ for each node $i \in V$
Ensure: Similarity matrix $D \in \mathbb{R}^{n \times n}$, where D_{ij} denotes similarity between nodes i and j

```
Initialize matrix D ← 0^{n×n}
for each node i ∈ V do
    for each node j ∈ V do
        if i = j then
            D_ij ← 1.0
        else if (i, j) ∈ E then
            Let N_i ← N(i), N_j ← N(j)
            Compute intersection: I ← N_i ∩ N_j
            Compute union (or denominator): d_i ← |N_i|
            D_ij ← (|I|+1) / d_i
        else
            D_ij ← 0
        end if
    end for
end for
return D
```

- U_j is the feature vector of node j;
- $|N(i)|$ is the number of non-zero similarity linkages for node i, which is its "soft neighborhood size";
- $(\cdot)$ means matrix multiplication or scaling a vector by its elements.

Alternatively, the entire matrix update can be expressed compactly as shown in Eq. 3:

$$U' = \text{Normalize}(S \cdot U) \tag{3}$$

This normalizes each row to consider the number of contributing neighbors. The cluster centroids $V \in \mathbb{R}^{c \times m}$ are then changed to the weighted average of the node representations. This is done by raising the fuzzy membership values to the fuzzifier parameter q, as illustrated in Eq. 4:

$$V_i = \frac{\sum_{k=1}^{n} (R_{ik})^q \cdot U_k}{\sum_{k=1}^{n} (R_{ik})^q} \tag{4}$$

The membership parameters for R_{ik} are modified according to the distance of the k-vector from each cluster centroid, demonstrated in Eq. 5.

$$R_{ik} = \left(\sum_{j=1}^{c} \left(\frac{||U_k - V_i||}{||U_k - V_j||} \right)^{\frac{2}{q-1}} \right)^{-1} \tag{5}$$

The process continues until convergence occurs, defined as the maximum change in R between two iterations falling below a predetermined threshold ϵ, or until the maximum iteration limit is attained.

D. Stage 4: Community Label Assignment and Optimization

In the final stage, we use the fuzzy membership matrix $R \in \mathbb{R}^{c\times n}$ to determine which community each node belongs to. Each of the entries of $R_{k,i}$ denotes the degree of membership of node i in community k. Eq. 6 indicates the procedure for acquiring an initial distinct partition by allocating each node to the community exhibiting the highest membership score in R. This phase creates a foundational framework in which each node belongs to a single community exclusively.

$$\text{label}i = \arg\max_{k} R_{ki} \tag{6}$$

We do a neighborhood-based optimization to allow for overlapping memberships and to better capture the complex local structure of the network. We find a community strength vector $C_i \in \mathbb{R}^c$ for each node i, where the k-th component $C_{i,k}$ is found using Eq. 7:

$$C_{i,k} = \frac{\sum_{j\in N(i)} S_{ij} \cdot \delta_{jk}}{\sum_{j\in N(i)} S_{ij}} \tag{7}$$

- S_{ij} is the similarity between node i and neighbor j,
- $\delta_{jk} = 1$ if node j is assigned to community k, and 0 otherwise,
- $N(i)$ denotes the set of neighbors of node i.

A node i is assigned to community k if its score $C_{i,k}$ Eq. 8 satisfies:

$$C_{i,k} \geq 0.9 \cdot \max_{l} C_{i,l} \tag{8}$$

The threshold rule ensures that nodes with comparable strength in more than one community are assigned to all of them, enabling controlled and meaningful overlaps. The iterative assignment process stabilizes the labels and ensures that overlapping community structures align well with both fuzzy membership scores and the local neighborhood structure.

This final optimization stage is crucial for transitioning from a distinct division to a nuanced, overlapping community detection outcome that effectively captures the complex relationships between nodes in the network.

E. Computational Complexity

The proposed technique's computing complexity is predominantly influenced by three stages: the formation of the neighborhood similarity matrix, the allocation of fuzzy memberships, and the enhancement of overlapping community memberships. The similarity matrix is generated by calculating local similarity, such as common neighbors, between pairs of connected nodes. The worst-case complexity is $\mathcal{O}(n^2)$ when all node pairs are compared; however, in reality, the computation is restricted to adjacent nodes, yielding a practical complexity of $\mathcal{O}(m \cdot d)$, where m represents the number of edges and d denotes the average node degree. The fuzzy clustering phase iteratively allocates community memberships with a Fuzzy C-Means-inspired update process, characterized by a complexity of $\mathcal{O}(n \cdot c \cdot t)$, where c denotes the number of communities and t represents the number of iterations. The refining phase modifies memberships according to neighborhood similarity and local consistency, exhibiting a complexity of $\mathcal{O}(n \cdot d \cdot c)$.

Thus, the proposed method attains a cumulative complexity of $\mathcal{O}(m\cdot d+n\cdot c\cdot(t+d))$, rendering it efficient and scalable for sparse real-world networks. Contrary to more computationally demanding methods such as OSLOM and BIGCLAM, our approach is efficient, interpretable, and particularly effective for large graphs, while still achieving competitive accuracy. For further understanding, Table II provides the precise comparison of computational complexities of different methods.

TABLE II. COMPUTATIONAL COMPLEXITY COMPARISON OF COMMUNITY DETECTION METHODS

Method	Complexity	Scalability	Key Operations
Proposed Method	$O(m \cdot d + n \cdot c \cdot (t + d))$	High	Local similarity computation, fuzzy membership updates, neighborhood refinement
OSLOM	$O(n \cdot d^{-1} \cdot \log n)$	Medium-Low	Local expansion and statistical significance testing
BIGCLAM	$O(k \cdot (n + m))$	High	Non-negative matrix factorization
GREESE	$O(k \cdot n \cdot d)$	Medium	Greedy seed expansion
LC	$O(t \cdot m)$	Very High	Label propagation, iterative voting

V. EXPERIMENTS AND RESULTS

The following section presents an experimental analysis of the proposed approach across networks of varying sizes and structural features to demonstrate its effectiveness in detecting overlapping communities. The evaluation is conducted by comparing it with numerous reputable overlapping community detection techniques, such as OSLOM [36], BIGCLAM [37], GREESE [23], and Label Clustering (LC) [38], respectively.

A. Evaluation Metric

We use two common measures for overlapping community detection, Overlapping Modularity (EQ) or Q_{ov}, and F-score, to measure how well the proposed approach works. The overlap modularity Q_{ov} is measured with no ground truth.

1) Overlapping modularity: EQ, or Q_{ov}, or overlapping modularity, is a new way to assess modularity that takes into account nodes that overlap. It evaluates the quality of the identified community structure by contrasting the number of edges within communities against a null model that permits nodes to belong to multiple communities. The EQ score is in Eq. 9 as:

$$EQ = \frac{1}{2m} \sum_{c} \sum_{i,j\in c} \left[A_{i,j} - \frac{k_i k_j}{2m} \right] \frac{1}{O_i O_j} \tag{9}$$

O_i is the number of communities that node i is a part of, k_i is the degree of node i, while m is the overall number of edges.

2) F-score: The F-score is a common way to assess how much the discovered communities overlap with the real ones. It is the harmonic mean of recall and precision, which is described for overlapping clustering as shown in Eq. 10:

$$F\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

B. Datasets

Table III outlines the datasets utilized in our experiments, with N denoting the number of nodes, E representing the number of edges, and k showing the average node degree. Our version utilizes Python's NetworkX package for graph preprocessing, whereas community assignment and similarity calculations are conducted using NumPy. The visualization of identified communities is executed with Matplotlib and the force-directed layout from igraph, facilitating qualitative analysis of community coherence and overlap. The comparison underscores the precision and computational efficacy of our methodology.

TABLE III. THE STATISTICS OF REAL-WORLD NETWORK DATASETS

Dataset	V	E	Avg. Degree	Description
Karate [39]	34	78	4.59	Zachary's Karate Club network representing friendships between 34 members of a university karate club.
Dolphins [40]	62	159	5.13	A network of 62 New Zealand bottlenose dolphins' frequent associations.
Football [41]	115	613	10.66	American college football team network with nodes representing teams and edges representing regular-season games.
Polbooks [42]	105	441	8.4	Amazon political book network during 2004 U.S. presidential election. Edges show often co-purchased books, and nodes depict books.
Jazz [43]	198	2742	27.7	Collaboration network of jazz musicians. An edge between two musicians means they have played in the same band.
Amazon [44]	334,863	925,872	5.53	A product co-purchasing network from Amazon, where nodes are products and edges link frequently co-purchased items.
Dblp [45]	317,080	1,049,866	6.62	A co-authorship network from DBLP, where nodes are authors and an edge indicates co-authorship of one or more papers.

C. Results and Discussion

In order to evaluate the effectiveness of the proposed method, we applied it to a set of widely used benchmark datasets and visualized the resulting community structures. The outcomes are displayed in Fig. 3, 4, 5, and 6, where each graph represents the community partitions generated by our algorithm. Distinct colors are used to indicate separate communities, while nodes that belong to more than one community are emphasized in red. These overlapping nodes highlight their dual or multiple affiliations, which often correspond to structurally important positions within the network, such as boundary spanners, information brokers, or connectors between clusters. The presence of such nodes underscores the inherently fuzzy and interconnected character of real-world community organization. By accurately identifying them, the proposed approach not only delineates clear community boundaries but also reveals the subtle overlaps and shared memberships that classical hard-clustering methods often fail to capture.

We evaluated the performance of the proposed method across several real-world network datasets and compared it to that of popular overlapping community detection methods, including OSLOM, BIGCLAM, GREESE, and LC. The outcomes show that the proposed method works better on networks of various sizes and typologies.

In smaller and medium-sized networks like Karate, Dolphins, and Jazz, the method proficiently identifies significant

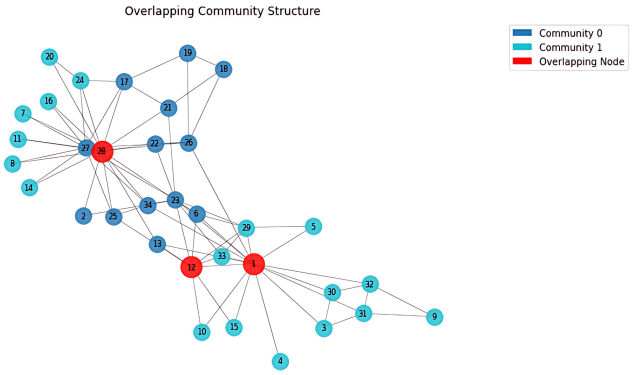


Fig. 3. Community visuals for Karate Club network datasets

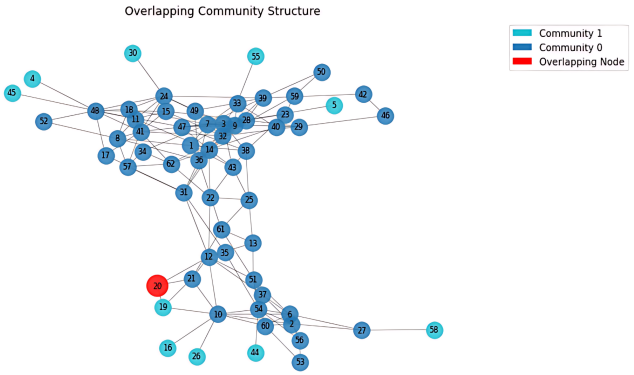


Fig. 4. Community visuals for Dolphins network datasets.

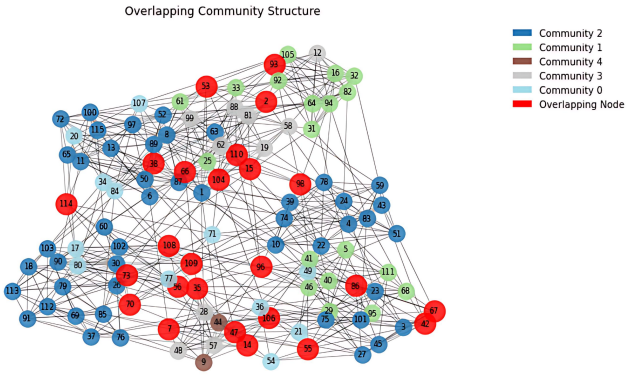


Fig. 5. Community visuals for Football network datasets.

overlapping groups, capturing significant dynamics that conventional methods often neglect. The fuzzy similarity-based approach enables the algorithm to assign nodes to various communities flexibly, capturing the fundamental uncertainty in community boundaries. While in large networks like dblp and Amazon, the proposed approach consistently yields dependable outcomes, demonstrating resilience and flexibility in handling high-dimensional data. Although certain baseline techniques may get slightly superior modularity in particular cases, our methodology demonstrates robust consistency across datasets,

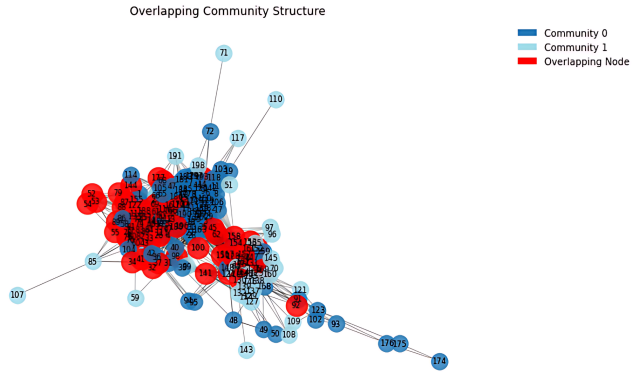


Fig. 6. Community visuals for Jazz network datasets.

particularly when community overlaps are significant.

Fig. 7 shows the comparison of overlapping modularity Q_{ov} results of the proposed method with four other state-of-the-art overlapping community detection methods. The proposed method attains an advantageous convergence between optimization and interpretability. It adeptly combines local topological characteristics with a systematic fuzzy clustering approach, providing a viable alternative to more complex or heavily parameterized algorithms.

Fig. 8 represents the F-score comparison on larger datasets. The F-score analysis on the DBLP and Amazon network datasets underscores the efficacy of the proposed method for precisely detecting overlapping groups. In contrast to many existing methods that either excessively cluster or inadequately identify significant overlaps, our methodology preserves a balanced framework by utilizing fuzzy membership scores and neighborhood-based refinement. In the DBLP dataset, the technique demonstrates consistent performance, closely agreeing with the ground truth while remaining competitive with more established algorithms. In the Amazon network, marked by thick overlaps and varied community structures, the technique exhibits constant precision and recall, indicating its adaptation to real-world complexities. The results indicate that the proposed approach can generate community assignments that are both precise and comprehensible, even in large networks with complex topologies.

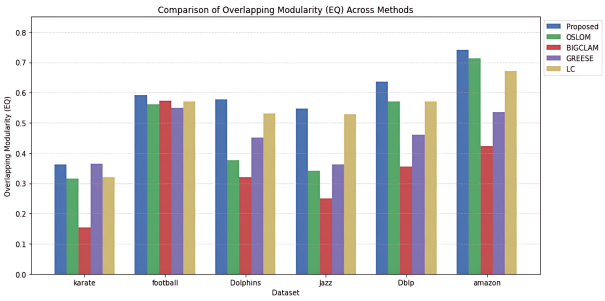


Fig. 7. Qov values of real-world datasets on different algorithms.

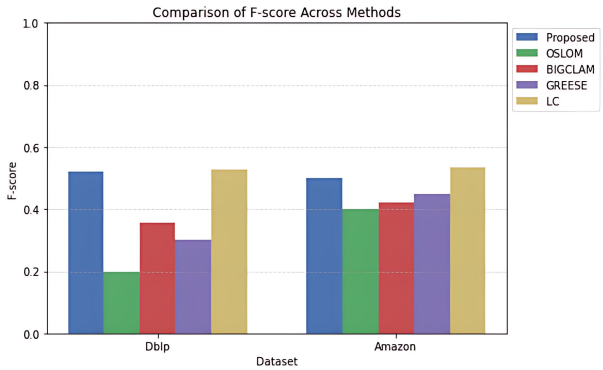


Fig. 8. F-score values of real-world datasets on different algorithms.

D. Execution Time

The bar chart in Fig. 9 depicts the execution time (in seconds) of different community discovery methods across four benchmark datasets: Karate, Football, Dolphins, and Jazz. OSLOM and BigClam consistently exhibit the lowest execution times, rendering them among the most computationally effective algorithms in our comparison. Conversely, LC exhibits the longest execution time across all datasets, signifying considerable computational expense.

The proposed method demonstrates moderate execution durations; it is often quicker than LC and GREESE, however marginally slower than OSLOM and BigClam. This is a judicious compromise between computing expense and efficacy, particularly given that the proposed approach emphasizes the precise identification of overlapping and structurally intricate communities. The Jazz dataset exhibits the longest execution durations among all approaches, presumably attributable to its greater size and edge density.

Although the proposed method may not be the most expedient, it ensures satisfactory execution times while offering enhanced community quality, particularly in contexts where overlapping detection is essential. This equilibrium indicates that the algorithm is sufficiently efficient for practical application, especially in small to medium-sized networks.

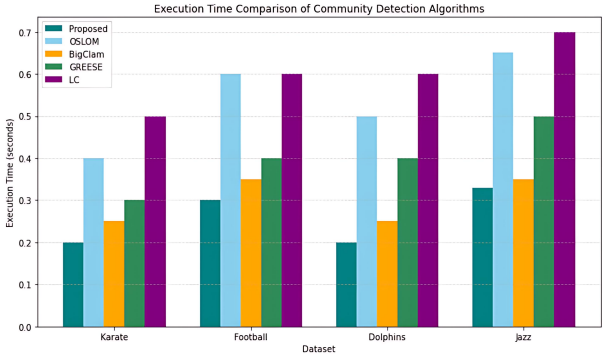


Fig. 9. Execution time of different algorithms compared with the proposed method.

E. Statistical Analysis

We used a non-parametric Friedman test on the EQ (Extended Modularity) values from many benchmark datasets to see if the variations in performance between community detection techniques were statistically significant. Fig. 10 demonstrates that the proposed method consistently outperforms the baseline approaches. The highest average EQ score is observed across all datasets, with the LC algorithm ranking second. The Friedman test demonstrates that the differences among the techniques are statistically significant, necessitating additional pairwise comparisons.

We used a Nemenyi post-hoc test to find out where the differences are, and the findings are shown in Fig. 11. This test evaluates the algorithms based on how well they do on average across datasets and shows if the differences are statistically significant. The proposed technique comes in first, followed by LC, while BIGCLAM always comes in last. The red line showing the Critical Difference (CD) shows that the difference in performance between the suggested technique and the lower-ranked algorithms is statistically significant. This analysis shows that our method is strong and competitive when it comes to finding overlapping communities with better quality outcomes.

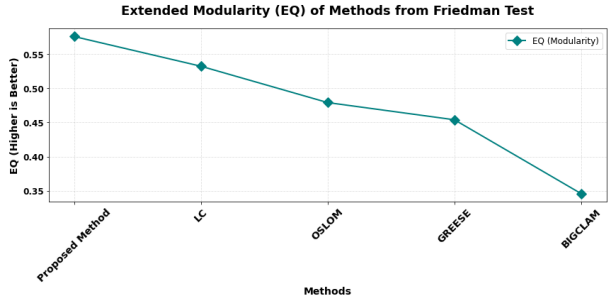


Fig. 10. Friedman statistical analysis results.

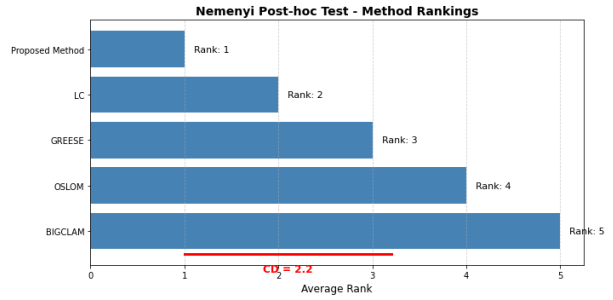


Fig. 11. Nemenyi post hoc test results.

VI. Application Domains of the Proposed Approach

The ability to detect overlapping communities with varying degrees of membership makes the proposed method applicable across numerous real-world domains where entities simultaneously participate in multiple functional groups [46], [47], [5]. Unlike hard clustering approaches, fuzzy membership and local topological similarity enable the representation of nuanced interactions, making the method suitable for both small-scale and large-scale systems. Table IV shows the primary application areas where the proposed approach can improve assessment, decision-making, and system optimization.

A. Social Networks

Identifying overlapping communities in social networks is crucial for recognizing users with numerous social circles. This helps platforms comprehend complex social interactions, improve tailored recommendations, and target content. Deeper insights into influence propagation and user engagement patterns enable more effective viral marketing efforts, community-driven promotions, and strategic network interventions using user relationships [48].

B. Recommender Systems

In the context of recommender systems, individuals frequently exhibit a wide array of interests spanning various domains, including music, films, and online shopping. Modeling diverse interest groups facilitates enhanced personalization and more precise recommendations [49].

C. Biological Networks

Biological networks, including protein-protein interaction networks, frequently display overlapping communities in which proteins contribute to various functional modules. Recognizing these overlaps can uncover biological entities with multiple functions and contribute to our comprehension of disease pathways [50], [51].

D. Information and Citation Networks

Documents within information and citation networks can encompass a variety of disciplines. The identification of overlapping communities has the potential to enhance the effectiveness of topic modeling and the efficiency of information retrieval [52].

E. Cybersecurity

User access logs and network traffic often reveal entities assuming multiple roles or connecting across domains. Traditional clustering may overlook overlapping patterns and subtle security risks. By capturing multi-role interactions, overlap-aware community detection enables analysts to study network structure and user behavior, helping identify unauthorized access, privilege escalation, and coordinated attacks. It also highlights advanced persistent threats, insider threats, and malware propagation, enhancing overall network security and monitoring [51].

F. Political and Behavioral Analysis

In the context of political and behavioral studies, it is common for individuals to possess multiple affiliations. This complexity underscores the importance of monitoring polarization and the interactions that occur across different groups [53].

TABLE IV. APPLICATIONS AND BENEFITS OF THE PROPOSED METHOD IN VARIOUS DOMAINS

Domain	Application	Advantage
Social Networks	Identifying users in multiple social circles	Enhanced recommendations and influence analysis
Recommender Systems	Modeling multi-interest user groups	Improved personalization and targeting
Biological Networks	Detecting overlapping functional gene/protein modules	Insights into multifunctional biological entities
Information/Citation Networks	Classifying documents or topics across disciplines	Better topic modeling and retrieval
Cybersecurity	Multi-role detection in access or traffic patterns	Improved intrusion detection and anomaly analysis
Political/Behavioral Analysis	Overlapping affiliations in opinion or behavior networks	Polarization tracking and behavioral segmentation
Marketing & Segmentation	Multi-interest customer clustering	Flexible and refined market profiling
Infrastructure/Transport	Identifying shared zones across transport routes	Route optimization and load balancing

G. Marketing and Infrastructure Systems

In the realm of infrastructure systems, especially within smart cities, overlapping community detection can be utilized to examine transportation networks, shared mobility facilities, and urban flow patterns. Vehicles, commuters, and logistics operators are some of the entities that typically use several routes or service regions that overlap. City planners and service providers can make better use of public transportation and shared mobility systems by finding these overlaps, which lets them improve traffic management, optimize transport timetables, and improve route allocation [54].

VII. CONCLUSION AND FUTURE WORK

This study presents a neighborhood similarity-based fuzzy clustering method for identifying overlapping communities in complex networks. Inspired by the fact that nodes in real-world systems frequently display partial membership across various communities, the proposed approach integrates local topological similarity into a fuzzy clustering framework to more accurately represent this fundamental ambiguity. The proposed approach establishes a similarity matrix based on shared neighbors and enhances community memberships by local optimization, achieving a balance among structural fidelity, interpretability, and computing efficiency. Through comprehensive analysis on standard benchmark datasets such as Karate, Dolphins, Polbooks, and so on, we established that our method competes effectively with renowned overlapping community detection techniques, for example, OSLOM, BIGCLAM, LC, and GREESE, frequently attaining enhanced outcomes both in terms of quality metrics like extended modularity and F-score. The method's architecture eliminates dependence on global heuristics or resource-intensive computations, rendering it scalable and accessible for extensive applications. This study emphasizes the significance of local structure in community inference and the necessity of adaptable, membership-aware frameworks for comprehending complicated networked systems, in addition to robust empirical findings.

Future research will investigate the incorporation of node properties and temporal dynamics in order to improve the model's relevance to changing and heterogeneous networks. We intend to integrate deep learning methodologies, specifically graph neural networks, to derive more expressive node representations that enhance community detection precision. Furthermore, we intend to examine the method's applicability in subsequent tasks, including link prediction, influence maximization, and anomaly detection.

ACKNOWLEDGMENT

This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia, under grant no. (GPIP: 282- 830-2024). The authors, therefore, acknowledge with thanks DSR for technical and financial support.

REFERENCES

[1] E. Egor, "Community detection on simplicial complexes: E. egor," *Applied Network Science*, vol. 10, no. 1, p. 30, 2025.

[2] K. Chi, H. Qu, and Z. Fu, "A novel approach for overlapping community detection in social networks based on the attraction," *Journal of Computational Science*, vol. 85, p. 102508, 2025.

[3] K. Kaliaperumal, M. L. Rathod, L. Raju, and J. Mahil, "Efficient overlapping community detection using mapreduce-based fuzzy c-means clustering on seed nodes," *IEIE Transactions on Smart Processing & Computing*, vol. 14, no. 1, pp. 1–10, 2025.

[4] Q. An, P. Wang, Y. Zeng, and Y. Dai, "Cooperative social network community partition: A data envelopment analysis approach," *Computers & Industrial Engineering*, vol. 172, p. 108658, 2022.

[5] F. R. Khawaja, Z. Zhang, Y. Memon, and A. Ullah, "Exploring community detection methods and their diverse applications in complex networks: a comprehensive review," *Social Network Analysis and Mining*, vol. 14, no. 1, p. 115, 2024.

[6] O. Doluca and K. Oğuz, "Apal: Adjacency propagation algorithm for overlapping community detection in biological networks," *Information Sciences*, vol. 579, pp. 574–590, 2021.

[7] M. Liu, J. Liu, Y. Dong, R. Mao, and E. Cambria, "Interest-driven community detection on attributed heterogeneous information networks," *Information Fusion*, vol. 111, p. 102525, 2024.

[8] M. Guerrero, F. Montoya, R. Baños, A. Alcayde, and C. Gil, "Community detection in national-scale high voltage transmission networks using genetic algorithms," *Advanced Engineering Informatics*, vol. 38, pp. 232–241, 2018.

[9] 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.

[10] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E*, vol. 70, no. 6, p. 066111, 2004.

[11] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, pp. 8271–8276, 2002.

[12] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *Journal of Graph Algorithms and Applications*, vol. 10, no. 2, pp. 191–218, 2006.

[13] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[14] Y. Liu, S. Liu, F. Yu, and X. Yang, "Link prediction algorithm based on the initial information contribution of nodes," *Information Sciences*, vol. 608, pp. 1591–1616, 2022.

[15] N. Arinik, V. Labatut, and R. Figueiredo, "Characterizing and comparing external measures for the assessment of cluster analysis and community detection," *IEEE Access*, vol. 9, pp. 20 255–20 276, 2021.

[16] H. Li, S. S. Nasab, H. Roghani, P. Roghani, M. Gheisari, C. Fernández-Campusano, A. A. Abbasi, and Z. Wu, "Lmfls: A new fast local multi-factor node scoring and label selection-based algorithm for community detection," *Chaos, Solitons & Fractals*, vol. 185, p. 115126, 2024.

[17] D. X. Li, P. Zhou, B. W. Zhao, X. R. Su, G. D. Li, J. Zhang, P. W. Hu, and L. Hu, "Biocaiv: An integrative webserver for motif-based clustering analysis and interactive visualization of biological networks," *BMC Bioinformatics*, vol. 24, p. 451, 2023.

[18] Y. Feng, H. Chen, T. Li, and C. Luo, "A novel community detection method based on whale optimization algorithm with evolutionary population," *Applied Intelligence*, vol. 50, pp. 2503–2522, 2020.

[19] M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig, "Community detection in networks: A multidisciplinary review," *Journal of Network and Computer Applications*, vol. 108, pp. 87–111, 2018.

[20] F. R. Khawaja, Z. Zhang, and A. Ullah, "Common-neighbor based overlapping community detection in complex networks," *Social Network Analysis and Mining*, vol. 15, no. 1, p. 61, 2025.

[21] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding statistically significant communities in networks," *PLOS ONE*, vol. 6, no. 4, p. e18961, 2011.

[22] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," in *Proceedings of the 6th ACM International Conference on Web Search and Data Mining (WSDM)*, 2013, pp. 587–596.

[23] K. Asmi, D. Lotfi, and A. Abarda, "The greedy coupled-seeds expansion method for the overlapping community detection in social networks," *Computing*, vol. 104, no. 2, pp. 295–313, 2022.

[24] K. Asmi, D. Lotfi, and M. El Marraki, "A new local algorithm for overlapping community detection based on clustering coefficient and common neighbor similarity," in *Proceedings of the ArabWIC 6th Annual International Conference Research Track*, 2019, pp. 1–6.

[25] N. Meghanathan, "A greedy algorithm for neighborhood overlap-based community detection," *Algorithms*, vol. 9, no. 1, p. 8, 2016.

[26] J. Liu, J. Guo, and Q. Li, "A local extended algorithm combined with degree and clustering coefficient to optimize overlapping community detection," *Complexity*, vol. 2021, no. 1, p. 7428927, 2021.

[27] Y. Li, J. He, Y. Wu, and R. Lv, "Overlapping community discovery method based on two expansions of seeds," *Symmetry*, vol. 13, no. 1, p. 18, 2020.

[28] Y. Song, Z. Zheng, Y. Shi, and B. Wang, "Glod: the local greedy expansion method for overlapping community detection in dynamic provenance networks," *Mathematics*, vol. 11, no. 15, p. 3284, 2023.

[29] J. Jia, P. Liu, X. Du, Y. Yao, and Z. Lei, "The overlapping community discovery algorithm based on the local interaction model," *Intelligent Data Analysis*, vol. 26, no. 1, pp. 153–171, 2022.

[30] A. Hasan and A. Kamal, "Lapefcm: overlapping community detection using laplacian eigenmaps and fuzzy c-means clustering," *International Journal of Information Technology*, vol. 14, no. 6, pp. 3133–3144, 2022.

[31] E. Pourabbasi, V. Majidnezhad, N. F. Veijouyeh, S. Taghavi Afshord, and Y. Jafari, "A novel intelligent fuzzy-ahp based evolutionary algorithm for detecting communities in complex networks," *Soft Computing*, vol. 28, no. 11, pp. 7251–7269, 2024.

[32] E. Jokar, M. Mosleh, and M. Kheyrandish, "Overlapping community detection in complex networks using fuzzy theory, balanced link density, and label propagation," *Expert Systems*, vol. 39, no. 5, p. e12921, 2022.

[33] A. Moradan, A. Draganov, D. Mottin, and I. Assent, "Ucode: Unified community detection with graph convolutional networks," *Machine Learning*, vol. 112, no. 12, pp. 5057–5080, 2023.

[34] M. N. Muttakin, M. I. Hossain, and M. S. Rahman, "Overlapping community detection using dynamic dilated aggregation in deep residual gcn," *arXiv preprint arXiv:2210.11174*, 2022.

[35] J. H. Patil, P. Potikas, W. B. Andreopoulos, and K. Potika, "Community detection using deep learning: Combining variational graph autoencoders with leiden and k-truss techniques," *Information*, vol. 15, no. 9, p. 568, 2024.

[36] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding statistically significant communities in networks," *PLoS ONE*, vol. 6, no. 4, p. e18961, 2011.

[37] J. Yang and J. McAuley, "Overlapping community detection at scale: A nonnegative matrix factorization approach," *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 587–596, 2013.

[38] D. Chen, M.-S. Shang, Z.-M. Lv, and Y. Fu, "Detecting overlapping communities of weighted networks via a local algorithm," *Physica A*, vol. 389, no. 19, pp. 4177–4187, 2010.

[39] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[40] D. Lusseau, "The emergence of coordinated communication in the bottlenose dolphin," *Trends in Cognitive Sciences*, vol. 7, no. 1, pp. 12–21, 2003.

[41] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[42] V. Krebs, "Mapping networks of terrorist cells," *Connections*, vol. 24, no. 3, pp. 43–52, 2003.

[43] P. M. Gleiser and L. Danon, "Community structure in jazz," *Advances in Complex Systems*, vol. 6, no. 4, pp. 565–573, 2003.

[44] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: Understanding rating dimensions with review text," in *Proceedings of the 7th ACM Conference on Recommender Systems*, 2013.

[45] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

[46] M. N. Al-Andoli, J. A. Alsayaydeh, I. M. Alwayle, C. K. N. C. K. Mohd, F. Abuhoureyah *et al.*, "Robust overlapping community detection in complex networks with graph convolutional networks and fuzzy c-means," *IEEE Access*, vol. 12, pp. 70 129–70 145, 2024.

[47] M. Naderipour, M. H. Fazel Zarandi, and S. Bastani, "Fuzzy community detection on the basis of similarities in structural/attribute in large-scale social networks," *Artificial Intelligence Review*, vol. 55, no. 2, pp. 1373–1407, 2022.

[48] X. Liu and Y. Zhang, "Overlapping community detection in social networks using a quantum-based genetic algorithm," *ACM Transactions on Knowledge Discovery from Data*, vol. 19, no. 4, pp. 1–22, 2025.

[49] N. Santhosh Kumar and S. Rajendran, "Recommender system using community detection and probabilistic matrix factorization," in *Proceedings of the International Conference on Artificial Intelligence*. Springer, 2023, pp. 123–130.

[50] P. Prokop and J. Novak, "Hierarchical overlapping community detection for weighted networks," in *Advances in Complex Systems*. Springer, 2023, pp. 157–170.

[51] P. Prokop and P. Drázdilová, "Overlapping community detection in weighted networks via graph hierarchical agglomerative clustering," *PLOS ONE*, vol. 19, no. 2, p. e0312596, 2024.

[52] V. Kuikka and X. Li, "Detecting overlapping communities based on influence spreading matrix and local maxima of a quality function," *Information Sciences*, vol. 612, pp. 1–15, 2024.

[53] Y. Zhang, X. Liu, and Z. Wang, "On the influence and political leaning of overlap between propaganda communities," *ACM Transactions on Internet Technology*, vol. 24, no. 3, pp. 1–20, 2024.

[54] X. Gu, J. Yang, and Q. Huang, "Overlapping community detection in vehicular social networks using graph attention autoencoder," *Sensors*, vol. 25, no. 8, p. 2601, 2025.

Vision-Based Autonomous Localization of Fall Protection Anchor Points on Transmission Towers Using Multi-View Geometric Perception

Chunqing Yang[1], Yu Peng[2], Jian Yu[3], Dongfeng Yu[4], Rui Liu[5], Jiahui Chen[6]
Leshan Power Supply Company, State Grid Sichuan Electric Power Company, Sichuan, 614000, China[1,2,3,4,5]
Electric Power Science Research Institute, State Grid Sichuan Electric Power Company, Sichuan, 610041, China[6]

***Abstract*—This paper presents the first systematic investigation into autonomous UAV-mounted fall protection lanyard (FPL) deployment for high-voltage transmission tower inspections, addressing a critical safety gap in the power industry where falls account for 34% of occupational fatalities. We propose a novel geometry-based solution to overcome three fundamental limitations of existing approaches: the isolated processing of UAV imagery without sensor fusion, unreliable 2D-to-3D spatial correspondence in anchor point detection, and the high annotation costs of supervised learning methods. Our technical contribution establishes a multi-view geometric perception framework that decomposes the FPL anchoring task into ridge line identification and optimal mounting point selection. The method first develops a spacial edge distance perception algorithm specifically for power inspection drones, which computes structural depth through plane-induced homography transformations of temporally matched line features. Subsequently, a mounting position planning algorithm integrates multiview geometric constraints with practical operational requirements including ladder proximity, diagonal steel avoidance, and temporal stability. Experimental validation on real-world power infrastructure data demonstrates superior performance compared to learning-based alternatives, achieving 10.98 MAE in positioning accuracy while maintaining 80ms processing efficiency for real-time operation. The proposed approach eliminates dependency on manual climbing and expert annotations, offering both theoretical advancements in stereo-environment perception for complex structures and immediate field applicability for safer power grid maintenance. This work represents the first formal proposal and comprehensive solution for autonomous FPL deployment in transmission tower inspection scenarios.**

Keywords*—Fall protection lanyard; transmission tower inspection; anchor point localization; multiview geometry; spacial edge distance perception; homography transformation

I. Introduction

Power transmission towers (PTT) are steel lattice structures supporting high-voltage power lines ($110kV$ to ultra-high voltage $\geq$ $1000kV$). These 25-215m tall structures require regular inspection, which still predominantly relies on hazardous manual climbing [1]. In China's power sector (2022), falls accounted for 40% of accidents and 34% of fatalities [2] (Fig. 2), making them the leading occupational hazard. While safety harnesses remain the primary protective measure, their effectiveness is compromised on PTTs that lack integrated fall arrest guide rails (Fig. 1). The absence of such safety infrastructure creates substantial challenges for workers attempting to properly anchor their FPLs to designated mounting points prior to ascending the structure. Therefore, the installation of FPLs urgently requires innovative technical solutions.

Recent years have witnessed the emergence of remotely piloted unmanned aerial vehicles (UAVs) for deploying purpose-engineered fall protection lanyard (FPL) anchoring devices to the apex of high-voltage PTTs. The manual approach inherently has scalability problems due to dependence on scarce specialists while introducing safety risks when operated by novices. Meanwhile, the advances in drone-assisted transmission tower inspection [3], [4], [5], [6], [7], [8], [9] have demonstrated potential for automated FPLs anchoring. To the best of our knowledge, we are the first to formally propose and systematically investigate the problem of autonomous UAV-mounted FPL deployment. While no prior studies have directly addressed this problem, related research on power line facility localization and defect detection has predominantly adopted image-level detector-based methodologies.

However, existing related approaches present three fundamental research gaps that hinder practical deployment: 1) The UAV video imagery are processed in isolation, failing to fully leverage the synergistic information potential between visual data and onboard sensor measurements. 2) Second, while some approaches employ deep learning-based object detection algorithms (e.g. YOLO series [10]) to identify potential anchor points in 2D image space, they fail to establish accurate 3D spatial correspondence, resulting in unreliable detection where visually suitable regions may not correspond to physically actionable locations on the tower structure, particularly when identified points are misaligned with the tower's front face. 3) Third, supervised deep learning requires extensive data annotation by power transmission tower inspection experts, leading to high labeling costs and limited scalability. The fundamental scientific challenge involves developing stereo-environment perception that simultaneously incorporates both the target structure's geometric characteristics and the drone's dynamic operational constraints.

To address these issues, we innovatively proposed an autonomous localization algorithm for FPL mounting points based on multi-view geometry. Drone-assisted FPL deployment involves ascending to operational altitude, horizontal positioning over the tower, and precise anchor point attachment. Our contribution is to guide the drone to automatically find the anchor point during the horizontal positioning stage. In this paper, the original anchoring task is decomposed into two sub-problems in our approach: 1) identifies the ridge line of

the tower's apex that is feasible to attach FPLs, and 2) selects the optimal anchoring point on the ridge line against various requirements. Therefore, we proposed a geometry model to identifies the ridge line of the tower's apex, then to screen the ridge points against specially designed objective functions, which combines geometric perception with practical operational constraints to achieve reliable performance in complex field environments. Experiments revealed that this technical approach enables autonomous drone-based FPL deployment, with potential value for both research and field applications.



Fig. 1. The quadcopter drone operates near the power transmission tower while carrying our specially designed mounting device for fall protection lanyards.

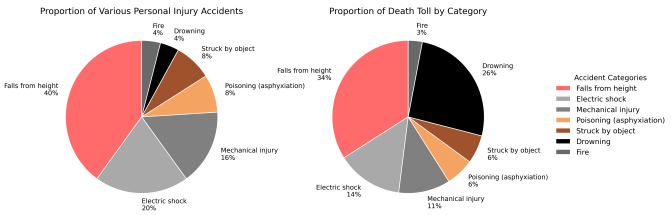


Fig. 2. Component pie-graph of personal injury accidents (left) and death toll (right).

The main contributions of this paper are as follows:

- A scene line segment distance perception algorithm specifically designed for power inspection drones.
- A mounting position planning algorithm for inspection fall protection lanyards.
- Experimental validation on real-world power infrastructure data, demonstrating the effectiveness and advancement of the proposed method.

The rest of this paper is organized as follows. Section II reviews related works. Section III describes the geometry model that links the linear features in images with the their depth in camera's view. Section IV explains the FPL anchoring point localization algorithm. Section V presents the experimental results. Section VI concludes the paper.

II. Related Works

A. Vision-Based UAV Power Line Inspection

Unmanned Aerial Vehicles (UAVs) have become vital for power line inspection [8], [11], offering efficient, high-precision monitoring across diverse environments. Recent developments integrate deep learning models—especially Convolutional Neural Networks (CNNs)—to detect anomalies like insulator fractures, vegetation encroachment, and conductor wear with improved accuracy [12]. These models benefit from large-scale annotated datasets and data augmentation techniques such as color space transformations and edge enhancement, which improve contrast between power line components and background clutter [12], [3]. Federated learning also shows promise for collaborative model training across UAV fleets while preserving data privacy [12], [5].

Despite these advances, spatial perception and geometric reconstruction remain key challenges [13]. Monocular SLAM frameworks like PTI-SLAM enable real-time mapping and localization, achieving trajectory RMSE of 0.1447 m [4]. However, they struggle with textureless surfaces and rapid rotations, leading to misalignment between image features and reconstructed scenes [14], [4], [15], [6]. In contrast, Multi-View Stereo (MVS) methods like DP-MVS offer higher geometric fidelity, achieving RMSE of 3.698 cm [16]. Yet, their computational intensity limits real-time deployment, highlighting a critical research gap: the lack of a unified framework that balances temporal responsiveness with spatial accuracy in complex structural environments. Addressing this gap is essential for reliable UAV-based inspections, particularly for identifying FPL mounting points on transmission towers.

B. Monocular SLAM for Transmission Tower

Monocular SLAM frameworks such as PTI-SLAM offer real-time localization and mapping capabilities for UAV-based power tower inspections [4], [9]. By combining direct and feature-based tracking with semantic filtering, PTI-SLAM achieves trajectory RMSE of 0.1447 m, outperforming GPS alignment methods [4]. Multi-frame fusion and statistical outlier removal further improve point cloud consistency, reducing noise while preserving structural accuracy [4]. However, these systems face key limitations in complex 3D environments.

A major challenge is the poor performance on textureless surfaces—common in metallic tower components—leading to unreliable feature detection and depth estimation errors [15]. Additionally, rapid rotational movements degrade tracking stability, lowering success rates to 6/10 under such conditions [4]. These issues cause misalignment between 2D image features and reconstructed 3D geometry, affecting tasks like defect classification and mounting point identification. Compared to MVS approaches like DP-MVS, which achieve RMSE of 3.698 cm, monocular SLAM lacks the geometric precision needed for detailed structural analysis [16]. Although PTI-SLAM processes frames efficiently at 81 ms per tracking step [4], its metric accuracy remains insufficient for high-fidelity inspection tasks. This highlights a key research gap: the absence of robust multi-view geometric analysis that maintains both real-time performance and spatial coherence in UAV-based power tower inspections.

C. Multi-View Stereo for Transmission Inspections

Multi-view stereo (MVS) methods provide significantly higher geometric accuracy than monocular SLAM, making them suitable for detailed 3D reconstruction of power towers [17]. Techniques like DP-MVS employ PatchMatch-based depth estimation and Delaunay meshing to preserve structural details, achieving RMSE as low as 3.698 cm [16]. Compared to traditional tools such as OpenMVS and COLMAP, DP-MVS

reduces processing time by up to 73%, enhancing efficiency for large-scale modeling [16]. Learning-based approaches like GC MVSNet++ further improve performance by embedding geometric constraints into training, cutting iteration counts by 50% [7], [18].

Despite these gains, MVS methods face challenges in real-time deployment due to high computational demands. They are also sensitive to textureless surfaces where feature correspondence is weak [19]. Progressive Prioritized MVS addresses efficiency concerns by selecting key viewpoints, reducing runtime by 42% while maintaining 98.7% completeness in reconstructed scenes [19]. However, integration into UAV inspection workflows remains limited due to hardware and synchronization requirements. Variable-baseline stereo setups using dual UAVs offer flexibility by adjusting stereo geometry dynamically, improving coverage and reducing occlusion [20]. Yet, they require precise pose estimation [21], which is difficult in GPS-denied or electromagnetically noisy environments. These limitations highlight a key research gap: the need for adaptive MVS frameworks that maintain high geometric fidelity while supporting real-time operation for critical tasks such as FPL mounting point detection.

III. Spacial Edge Perception Model

A. Coordinate system

This study employs a quadrotor UAV system for power transmission tower (PTT) inspection missions. The camera is mounted on the ventral side of the drone via a gimbal with controllable pitch angle. Since we focused on analyzing the relative spatial relationship between the PTT and the UAV, the camera coordinate system is selected as the reference frame for subsequent analysis. To comply with the convention in digital image processing where the y-axis of images points downward, this study defines an right-handed camera coordinate system as illustrated in Fig. 3. The x-axis corresponds to the lateral direction of the UAV, the z-axis aligns with the optical axis toward the observation target. Considering the continuous motion of the UAV in flight, the captured images exhibit dynamic variations across different timestamps. Therefore, the camera coordinate system at time t is adopted as the reference frame fro multi-view geometry analysis.

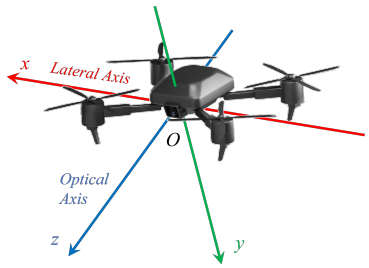


Fig. 3. Coordinate system of the inspection drone's vision.

B. Plane Induced Homography Transformation

Although a PTT appears visually complex, its components primarily consist of planar truss structures. Thus, we first analyze the image flow induced by spatial points on a plane. According to the findings in [22], a moving camera induces a homography transformation in the camera's view of points on a spatial plane. Specifically, let $\mathbf{x}_t \in \mathbb{P}^2$ and $\mathbf{x}_{t+1} \in \mathbb{P}^2$ denote the homogeneous coordinates of a feature point projected at time t and $t+1$, respectively. They satisfy:

$$\mathbf{x}_{t+1} = \mathbf{H}\mathbf{x}_t, \tag{1}$$

where $\mathbf{H} \in \mathbb{R}^{3\times3}$ represents the homography matrix. Moreover, we prove that the homography $\mathbf{H}$ satisfies the following proposition:

Proposition III.1. *(Homography): Let the camera coordinate system at time t be the reference frame. If the camera motion from t to $t+1$ is described by a translation vector $\mathbf{c} \in \mathbb{R}^3$ and a rotation matrix $\mathbf{R} \in SO(3)$, and the spatial plane is parameterized by its intercept d and normal vector $\mathbf{n}$ (with the plane equation $\mathbf{n}^T\mathbf{x} + d = 0$), then the homography transformation $\mathbf{H} \in \mathbb{R}^{3\times3}$ induced by this plane in the camera observation can be expressed as:*

$$\mathbf{H} = \mathbf{K}\mathbf{R}\left(\mathbf{I} + \frac{1}{d}\mathbf{c}\mathbf{n}^T\right)\mathbf{K}^{-1}, \tag{2}$$

where $\mathbf{K}$ is the camera's intrinsic parameter matrix, and $\mathbf{I}$ is the identity matrix.

Proof: See Appendix A.

C. Line Depth Approximation From Motion

The Real-Time Kinematics (RTK) system mounted on the drone provides onboard sensors to capture the motion parameters of the camera. Taking the camera frame at time t as the reference, both the translation vector $\mathbf{c} \in \mathbb{R}^3$ and the rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$ at time $t+1$ can be known. Additionally, the camera intrinsic matrix $\mathbf{K}$ can be acquired in advance. Thus, in (2), only the plane parameters (i.e. the intercept d and normal vector $\mathbf{n}$) remain unknown. Since that the camera consistently points toward the monitoring target (i.e. transmission towers), it is reasonable to assume that the target's surfaces are approximately perpendicular to the optical axis, i.e. $\mathbf{n} \approx (0,0,1)^T$. Under this assumption, (2) can be simplified to $\mathbf{H} = \mathbf{K}\mathbf{R}\hat{\mathbf{H}}\mathbf{K}^{-1}$, where

$$\hat{\mathbf{H}} = \mathbf{I} + \frac{1}{d}[\mathbf{0},\mathbf{0},\mathbf{c}] \tag{3}$$

Under this assumption, d represents both: (i) the distance from the camera's *optical center* to a quasi-planar surface (approximately parallel to the *principal plane*), and (ii) the orthogonal distance from the surface point to the principal plane. Furthermore, $\mathbf{H}$ is a conjugated transformation of $\mathbf{R}\hat{\mathbf{H}}$, where $\hat{\mathbf{H}}$ is a homology transformation [22]. If the camera's motion, i.e. $\mathbf{R}$ and $\mathbf{c}$ are provided by RTK, it is possible to obtain the d that supplement the target's depth in image.

Although (3) provides the foundation for depth analysis, the visual complexity of PTT scenarios makes point-based analysis unsuitable.

Given that structural edges constitute the most salient features of PTTs (Fig. 6), the depth should be analyzed based on line segments under a homography transformation.

Specifically, if a line segment at time t, denoted as $\mathbf{p} = (\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{P}^2 \times \mathbb{P}^2$, is mapped via homography $\mathbf{H}$ to a segment $\mathbf{q} = (\mathbf{x}'_1, \mathbf{x}'_2) \in \mathbb{P}^2 \times \mathbb{P}^2$ at time $t+1$, then any point $\mathbf{x}$ on $\mathbf{p}$ transformed by $\mathbf{H}$ must satisfy the line equation of $\mathbf{q}$. The line's coefficients are the cross product of the homogeneous endpoints, $\mathbf{x}'_1 \times \mathbf{x}'_2$.

This approach leverages the *duality between points and lines* in parametric space. This geometric constraints enables a line-based solution of d, leading to the following proposition:

Proposition III.2. *Let the camera coordinate system at time t be the reference frame. If a line segment at time t, denoted as $\mathbf{p} = (\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{P}^2 \times \mathbb{P}^2$, is mapped to the segment $\mathbf{q} = (\mathbf{x}'_1, \mathbf{x}'_2)$ at time $t+1$ via the homography transformation $\mathbf{H} \in \mathbb{R}^{3\times3}$ described in Formula (1), then for any point $\mathbf{x} \in \mathbb{P}^2$ on $\mathbf{p}$, its distance d to the principal plane of the camera coordinate system is given by:*

$$d = -\frac{\mathbf{c}^T\mathbf{R}^T\mathbf{K}^T(\mathbf{x}'_1 \times \mathbf{x}'_2)}{\mathbf{x}^T\mathbf{K}^{-T}\mathbf{R}^T\mathbf{K}^T(\mathbf{x}'_1 \times \mathbf{x}'_2)},$$

where, $\mathbf{c} \in \mathbb{R}^3$ and $\mathbf{R} \in SO(3)$ are the camera translation vector and rotation matrix from t to $t+1$, respectively; $\mathbf{K}$ is the camera intrinsic matrix; $\mathbf{x}'_1 \times \mathbf{x}'_2$ denotes the cross product of the homogeneous coordinates of the endpoints, defining the line equation of $\mathbf{q}$.

Proof: Given that $\mathbf{H} = \mathbf{K}\mathbf{R}\hat{\mathbf{H}}\mathbf{K}^{-1}$, for any point $\mathbf{x}$ on line segment $\mathbf{p}$, its corresponding point at time $t+1$, $\mathbf{x}' = \mathbf{K}\mathbf{R}\hat{\mathbf{H}}\mathbf{K}^{-1}\mathbf{x}$, must lie on the line defined by $\mathbf{q} = (\mathbf{x}'_1, \mathbf{x}'_2)$, i.e., $(\mathbf{x}'_1 \times \mathbf{x}'_2)^T\mathbf{K}\mathbf{R}\hat{\mathbf{H}}\mathbf{K}^{-1}\mathbf{x} = 0$. With $\mathbf{R}\mathbf{K}$ known, let $\mathbf{u}' = \mathbf{R}^T\mathbf{K}^T(\mathbf{x}'_1 \times \mathbf{x}'_2)$ and $\mathbf{u} = \mathbf{K}^{-1}\mathbf{x}$, we derive:

$$\mathbf{u}'^T\hat{\mathbf{H}}\mathbf{u} = 0. \tag{4}$$

Substituting (3), the equation expands to:

$$\mathbf{u}'^T\left(\mathbf{I} + \frac{1}{d}[\mathbf{0}, \mathbf{0}, \mathbf{c}]\right)\mathbf{u} = 0. \tag{5}$$

Solving for d from (5) yields:

$$d = -u_3\frac{\mathbf{c}^T\mathbf{u}'}{\mathbf{u}^T\mathbf{u}'}. \tag{6}$$

Since $u_3 = [0, 0, 1][x, y, 1]^T = 1$, the final distance formula is:

$$d(\mathbf{p}, \mathbf{q}) = -\frac{\mathbf{c}^T\mathbf{u}'}{\mathbf{u}^T\mathbf{u}'} = -\frac{\mathbf{c}^T\mathbf{R}^T\mathbf{K}^T(\mathbf{x}'_1 \times \mathbf{x}'_2)}{\mathbf{x}^T\mathbf{K}^{-T}\mathbf{R}^T\mathbf{K}^T(\mathbf{x}'_1 \times \mathbf{x}'_2)} \tag{7}$$

■

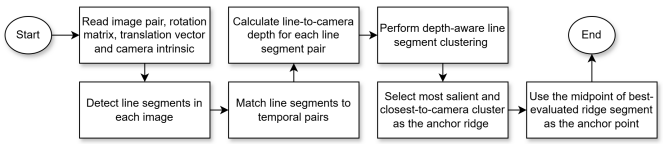


Fig. 4. Workflow of determining anchor ridge that contains fall arrest lanyard.

IV. Planning Safety Rope Anchorage Points

A. Determining Anchor Ridge of Fall Arrest Lanyard

Following the "anchor high, access low" principle for FPL [23], we define the anchor ridge as the load-bearing horizontal edge (the intersection of horizontal and vertical steel structures) at the apex of power transmission towers, which inherently includes appropriate anchor points. In UAV vision, the anchor ridge is the nearest ridge structure relative to the camera. Building upon the spacial line perception model (Section III-C), this subsection presents the detection of the anchor ridge, thus reducing the anchor point positioning problem to screen points along the one-dimensional anchor ridge (discussed in Section IV-B). The core workflow is outlined in Fig. 4, which contains key steps as follows:

1) Line segment detection: First, a line detection algorithm extracts multiple line segments from the current frame at time t, forming a set denoted as $\mathcal{S}^t = \{\mathbf{q}_1, \cdots, \mathbf{q}_m\}$. For all line segment $\mathbf{q} \in \mathcal{S}^t$, their endpoint $\mathbf{q} = (\mathbf{x}'_1, \mathbf{x}'_2)$ are in homogeneous coordinates, $\mathbf{x}'_1, \mathbf{x}'_2 \in \mathbb{P}^2$. Similarly, the prior frame's detection results at $t-1$ are denoted as $\mathcal{S}^{t-1} = \{\mathbf{p}_1, \cdots, \mathbf{p}_n\}$, where $\forall \mathbf{p} \in \mathcal{S}^{t-1}, \mathbf{p} = (\mathbf{x}_1, \mathbf{x}_2)$ and $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{P}^2$. Since lines are salient features in power tower scenes, various line segment detectors (e.g. Line Segment Detector (LSD) [24], Fast Line Detector (FLD) [25], Edge Drawing (ED) [26] or ELSED [27]) are applicable.

2) Temporal line segment matching: Given the two sets of line segments $\mathcal{S}^{t-1}$ and $\mathcal{S}^t$, the corresponding line segments that associate with the same structure on the transmission tower are expected to be paired, yielding a collection of matched line segment tuples.

$$\mathcal{P} = \left\{(\mathbf{p}, \mathbf{q}) | \mathbf{p} \in \mathcal{S}^{t-1}, \mathbf{q} = \arg\max_{\mathbf{q}' \in \mathcal{S}^t} p(\mathbf{p}, \mathbf{q}'), p(\mathbf{p}, \mathbf{q}) > 0\right\}. \tag{8}$$

where, $p(\mathbf{p}, \mathbf{q})$ denotes the probability of line segment $\mathbf{p}$ corresponding to segment $\mathbf{q}$. This can be achieved by line segment's invariant feature extraction and pairwise comparison. To ultimately achieve efficient operation on drone platforms, we adopted the Line Band Descriptor (LBD) [28] to extract stable features from line segments, followed by rapid line matching using KNN-Match algorithm [29]. These algorithms are integrated into the OpenCV library, enabling convenient and efficient implementation.

3) Depth computation and tuple construction: For each line segment pair $(\mathbf{p}, \mathbf{q}) \in \mathcal{P}$, its corresponding edge in the three-dimensional space has a depth d that is the distance from the edge to the camera's principal plane at time t. Eq. (7) is used to calculated the depth d, therefore yielding a set of triplets,

$$\mathcal{R} = \{(\mathbf{p}, d_{\mathbf{p}}) \mid (\mathbf{p}, \mathbf{q}) \in \mathcal{P}, d_{\mathbf{pq}} = d(\mathbf{p}, \mathbf{q})\}. \tag{9}$$

The distance $d(\mathbf{p}, \mathbf{q})$, computed via the previously established spacial edge perception model, reflects the geometric relationship between segment $\mathbf{p}$ and the principal plane.

4) Ridge detection by line segment clustering: Since the anchor ridge represents a prominent geometric feature composed of multiple steel bars, we employ depth-aware agglomerative hierarchical clustering (AHC) [30] to group segmented lines $\mathcal{R}$, generating large-scale linear structures $\mathcal{T} = \{\mathcal{O}_1, \cdots, \mathcal{O}_K\}$ through the aggregation of smaller segments, where $\mathcal{O}_i = \{\mathbf{p}, \cdots\}$ for $i \in \{1, \cdots, K\}$ is the i-th cluster of line segments. The depth-aware is achieved by introducing the depth as the extra feature. Then the anchor ridge is intrinsically retained within the clustered results,

$$\mathcal{T} = \texttt{clustering}_{\text{AHC}}(\mathcal{Q}). \tag{10}$$

To identify the anchor ridge, we select the clustered results with both smaller depth values and greater total segment lengths. Therefore, we have the anchor ridge $\hat{\Omega}$ chosen as

$$\hat{\Omega} = \arg\max_{\Omega \in \mathcal{T}} \sum_{(\mathbf{p}, d_{\mathbf{p}}) \in \Omega} \frac{\|\mathbf{p}\|}{d_{\mathbf{p}}}. \tag{11}$$

The cluster $\hat{\Omega}$ essentially represents the tower's nearest ridge to the camera. If the drone is in proper position, the meaningfulness of $\hat{\Omega}$ if ensured by the geometric stability of the truss structure.

Upon identifying the anchor ridge, the optimal mounting point can be determined through a search along this ridge. The localization algorithm for this process is described in detail in the following section.

B. Locating the Anchor Point for the Safety Lanyard

According to the analysis in Section IV-A, the optimal cluster $\hat{\Omega}$ represents a set of nearest salient edges on the transmission tower related to the drone (*mounting ridge lines*).

This subsection presents the anchor point locating algorithm, which aims to select the midpoint of the most suitable edge $\hat{\mathbf{p}}$ from the cluster $\hat{\Omega}$. To identify the most suitable anchor edge $\hat{\mathbf{p}} \in \hat{\Omega}$, we proposed a multi-objective cost function that evaluates the effectiveness of potential edges,

$$\hat{\mathbf{p}} = \arg\min_{\mathbf{p} \in \hat{\Omega}} \left[\psi(\mathbf{p}) + \lambda n(\mathbf{p})) + \omega \phi(\mathbf{p})\right]. \tag{12}$$

where, $\psi(\mathbf{p})$, $n(\mathbf{p}))$ and $\phi(\mathbf{p})$ are functions that evaluate different suitableness of the edge $\mathbf{p}$, and λ and ω are their weight factors.

Specifically, the cost function comprises the following components: 1) *Ladder proximity*: Although multiple suitable mounting points exist along the anchor ridge, we recommend prioritizing positions near the climbing ladder among the available options. Therefore, we model this evaluation using the ψ function. $\psi(\mathbf{p}) = \|\texttt{center}(\mathbf{p}) - \mathbf{x}_{\text{ladder}}\|$ quantifies the horizontal distance between the line segment's midpoint and climbing ladder, prioritizing segments closer to the ladder. Since $\mathbf{p} = (\mathbf{x}_1, \mathbf{x}_2)$, then $\texttt{center}(\mathbf{p}) = (\lfloor \mathbf{x}_1 \rfloor + \lfloor \mathbf{x}_2 \rfloor)/2$ with operator $\lfloor \cdot \rfloor$ representing homogeneous-to-Euclidean coordinate conversion. The $\mathbf{x}_{\text{ladder}} \in \mathbb{R}^2$ is calculated by object detection methods in advance or manually given by the operator. 2) *Vertical frame avoidance*: Considering that the anchor point needs to avoid interference from adjacent vertical steel frames, we define function $n(\mathbf{p})$ to quantitatively evaluate this condition. $n(\mathbf{p}) = |\{\mathbf{q}| \ \mathbf{q} \in \mathcal{R}, \mathbf{q} \notin \hat{\Omega}, e(\mathbf{p}, \mathbf{q}) > 0\}|$ counts adjacent interfering diagonal steel-bars, so as to avoid anchoring in structurally complex areas. λ is a penalty coefficient requiring adjustment. 3) *Length Suitability*: Since steel structures suitable for mounting must possess appropriate lengths (neither too short nor too long), we use the median length of line segments in $\hat{\Omega}$ as the reference length, and then evaluate the deviation between the length of the target segment $\mathbf{p}$ and this reference value. Therefore, $\phi(\mathbf{p}) = (\|\mathbf{p}\| - l_{\text{med}})$, where $l_{\text{med}} = \texttt{median}(\{\|\mathbf{q}\| | \mathbf{q} \in \hat{\Omega}\})^2$ is the median length of line segments in the anchor ridge $\hat{\Omega}$. Finally, the optimal anchor point $\hat{\mathbf{x}}$ is obtained by taking the middle point of $\hat{\mathbf{p}}$, namely, $\hat{\mathbf{x}} = \texttt{center}(\hat{\mathbf{p}})$.

[bt] **Input:** Input parameters $\mathfrak{I}_{t-1}$, $\mathfrak{I}_t$, $\mathbf{R}$, $\mathbf{c}$, $\mathbf{K}$

Output: Result $\hat{\mathbf{p}}$

if $\mathcal{S}_{t-1}$ *not exist* **then**
  $\mathcal{S}_{t-1} \leftarrow \texttt{detect_line}(\mathfrak{I}_{t-1})$
end

$\mathcal{S}_t \leftarrow \texttt{detect_line}(\mathfrak{I}_t)$ $\mathcal{P} \leftarrow \texttt{match_line_pairs}(\mathcal{S}_{t-1}, \mathcal{S}_t)$; ; // Eq.8

$\mathcal{R} \leftarrow \texttt{calculate_depth}(\mathcal{P}, \mathbf{R}, \mathbf{c}, \mathbf{K})$; ; // Eq.9

$\mathcal{T} \leftarrow \texttt{line_clustering}(\mathcal{R})$; ; // Eq.10

$\hat{\Omega} \leftarrow \texttt{select_cluster}(\mathcal{T})$; ; // Eq.11

$\hat{\mathbf{p}} \leftarrow \texttt{select_point}(\hat{\Omega})$; ; // Eq.12

return $\hat{\mathbf{p}}$

V. Experimental Results

A. Dataset

Since this study involves specialized experiments in power system applications, our investigation found no relevant publicly available datasets. Therefore, we independently collected experimental data (using devices show in Fig. 5) and manually annotated the optimal mounting positions for fall protection lanyard (FPL), constructing the TOWER dataset. The TOWER dataset originates from real-world power tower inspection tasks, captured by high-resolution cameras mounted on unmanned aerial vehicles (UAVs). During data collection, the UAV's real-time kinematic (RTK) positioning information was synchronously recorded to ensure spatial accuracy. The dataset consists of 2,380 sets of color images, with each set containing two consecutive frames at a resolution of 3840×2160 pixels, providing clear details of the towers and surrounding environments. The corresponding RTK data

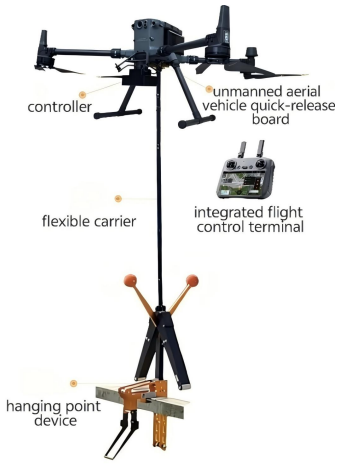


Fig. 5. The power inspection drone and fall protection lanyard equipment used to collect experimental data in this experiment.



Fig. 6. Sample images from the proposed TOWER dataset.

for each frame is stored in JSON format, facilitating image-geolocation alignment. The dataset is divided into training and testing subsets, with the test set containing 620 image pairs and the remaining 5,400 image pairs allocated for training. A key focus of the dataset is the optimal mounting positions for safety ropes. Some images include manual annotations marking the best attachment points, recording both image coordinates and ladder orientation. These annotations support the evaluation of detection algorithms for localization accuracy. Images without viable FPL mounting points are labeled as empty, serving as negative samples to enhance model generalization. The design and annotation methodology of the TOWER dataset provide robust data support for automated power facility inspection and maintenance, while also offering valuable practical insights for computer vision applications in the power industry. Our experimental dataset is planned for future public release.

TABLE I. DATASET DESCRIPTION

Name	Description
Dataset $\mathcal{D} = \{(I_i^0, I_i^1, R_i, t_i, K_i)\}_{i=1}^{N}$	Total $N = 2380$ items
Image Pair $I_0, I_1 \in \{0, \cdots, 255\}^{H \times W \times 3}$	(W,H) = (3840, 2160), 8-bits JPG
Rotation Matrix $\mathbf{R}_i \in \mathrm{SO}(3)$	float type, Json format
Translation Vector $\mathbf{c}_i \in \mathbb{R}^3$	float type, Json format
Camera Intrinsic $\mathbf{K}_i \in \mathbb{R}^{3\times3}$	float type, Json format

B. Hyper-parameter Setting

The experiments were conducted on a PC hardware platform equipped with an i7 processor and 32 GB of RAM to ensure efficient program execution. The software environment utilized Python 3.12, with key dependencies including OpenCV 4.9.0 for line detection algorithms, NumPy 1.26.4 for numerical computation, and Scikit-learn 1.6.1 for clustering algorithm implementation. The results obtained under this configuration effectively reflect the program's performance on typical computing systems, ensuring generalizability and reference value.

The line segment detector in our experiments was the ELSED [27].

The experimental parameters were configured as follows: $T_{\text{ori}} = 0.95$, $T_{\text{len}} = 0.8$, $D_{\text{pos}} = 80$ pixels $\lambda = 5.5$, $\omega = 0.9$.

C. Quantitative Experimental Analysis

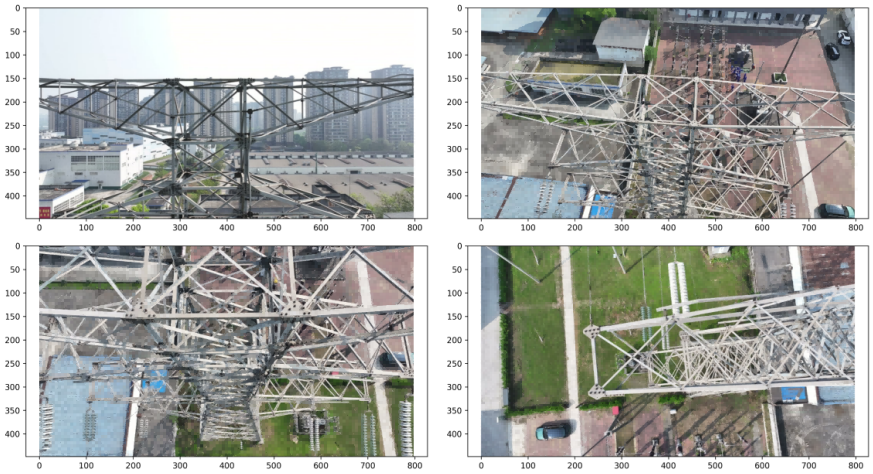
The experimental results demonstrate the effectiveness of our proposed algorithm in accurately identifying anchor ridges and locating optimal mounting points for fall protection lanyards (FPLs), as shown in Fig. 7. The original images are given in Fig. 7a. As illustrated in Fig. 7b, the depth-aware line segment matching successfully isolates structural edges on the transmission tower, with line color represents the depth values, e.g. closer ridges (magenta) clearly distinguished from background clutter (cyan). This enables precise ridge extraction, as shown in Fig. 7c, where the algorithm robustly detects the primary anchor ridges (red lines) and anchor points (black cross) despite complex steel truss interference.

Quantitative evaluation reveals strong alignment between our automated detection and manual annotations (ground-truth red circles). The mean absolute error (MAE) of anchor point positioning is 10.98 pixels, equivalent to sub-optimal positions at typical inspection distances, meeting industrial safety requirements. Our methodology's advantages primarily derive from geometric consistency. The plane-induced homography (Section III-B) maintains spatial coherence in ridge identification, minimizing erroneous detections from diagonal beam structures.

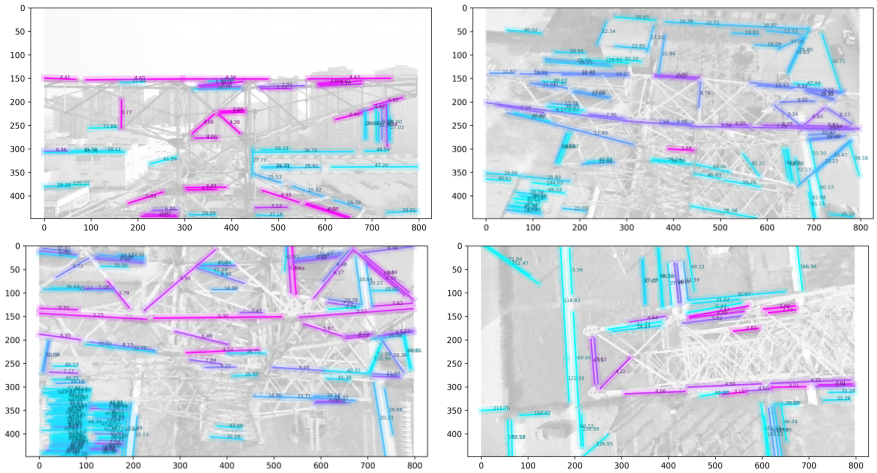
These results validate that our geometry-driven approach outperforms learning-based methods (Table II) in reliability and precision, particularly in preserving spatial relationships critical for FPL deployment safety. The minor deviations from manual markings primarily occur in incorrect line-pair match, highlighting opportunities for future refinement through robust line segment processing.

D. Positioning Accuracy and Efficiency Analysis

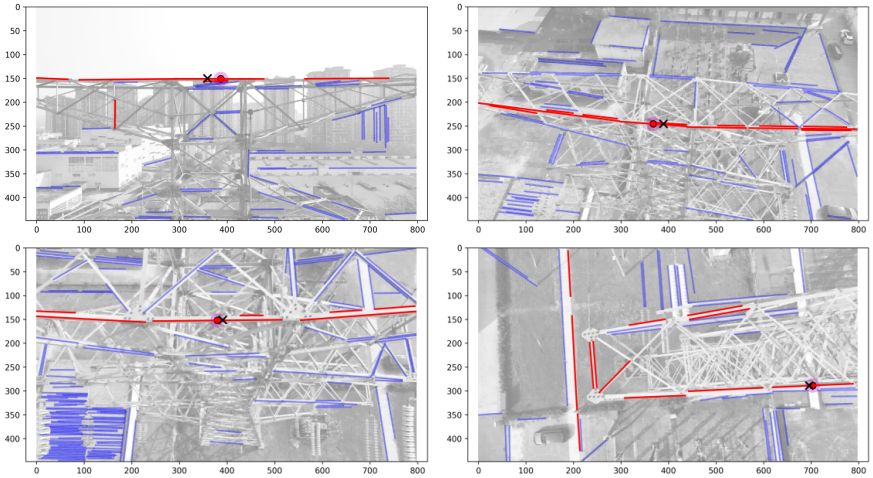
We compared the proposed algorithm with supervised learning-based neural network object detection algorithms.



(a) Original images



(b) Matched line segments with their depth, which is visualized by colormap that cyan indicates far and magenta represents near.



(c) Result of ridge detection (red) and anchoring point localization (cross). The red circle is ground-truth.

Fig. 7. Quantitative analysis of proposed method on various scenarios.

The experimental results are shown in Table I. Vision-based neural models were fine-tuned using our dataset, while the multimodal large model (e.g. DeepSeek V3) was directly invoked through prompt engineering. The analysis of the

experimental results indicates that ResNet30+RE and Our method achieve the best overall performance. Our method demonstrates the strongest robustness with the lowest MAE (10.98) and MedAE (11.02), while also achieving the highest R^2 (0.9725), indicating superior prediction accuracy and model fitting. ResNet30+RE follows closely, with slightly better MSE (265.57) and MAE (11.06) compared to ResNet18+RE, and its R^2 (0.9638) is close to that of Our method, demonstrating strong stability. In contrast, YOLOv5 and DeepSeekV3 exhibit higher MSE (288.72, 420.31), making them more sensitive to outliers, while Template Match is clearly unsuitable due to its excessively large error (MSE=7867.52).

Overall, while ResNet30+RE remains a supervised learning model, our method is unsupervised, eliminating the need for labeled data. This avoids the high costs and potential biases associated with data annotation in supervised learning, making it more advantageous when handling complex or unlabeled datasets.

TABLE II. PERFORMANCE COMPARISON OF DIFFERENT METHODS

Method	MSE	MAE	R^2	MedAE
Template Match	7867.52	86.47	0.6554	90.43
Yolov5	288.72	16.45	0.9107	18.74
ResNet18+RE	276.54	12.12	0.9535	13.16
ResNet30+RE	265.57	11.06	0.9638	12.93
DeepSeekV3	420.31	17.88	0.8989	19.22
Our	268.03	10.98	0.9725	11.02

E. Algorithm Efficiency Experiments

Due to the unique operational environment of UAVs, their environmental perception requires high dynamic responsiveness, making the computational efficiency of UAV algorithms critical. Therefore, we conducted experiments on time efficiency, with comparative results shown in Fig. 8.

The figure illustrates the performance of the proposed algorithm in processing varying numbers of line segments. The x-axis represents different levels of line segment quantities, including the most reliable top 100, 200, and 400 lines (top100, top200, and top400), while the y-axis displays the processing time in milliseconds. Performance at each level is measured using both the mean and standard deviation. As observed, the average processing time increases gradually with the number of line segments. Specifically, the mean processing time for top100 segments is approximately 80 ms, with a small standard deviation, indicating stable performance. These results demonstrate that the proposed algorithm can achieve real-time or near-real-time video processing by controlling the number of line segments to be processed, thereby meeting industrial requirements.

F. Limitations

Through our numerical experiments, we identified several limitations in the current system:

1) Line segment extraction and matching robustness issues: The performance of line detection and temporal matching remains sensitive to environmental variations, such as changes in lighting conditions and complex backgrounds. In extreme cases, this can lead to complete failure in segment extraction or incorrect matching, compromising the system's reliability.

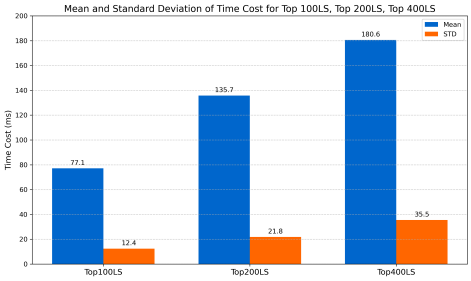


Fig. 8. Performance of the proposed algorithm in processing varying numbers of line segments.

2) Depth estimation accuracy limitations: While the proposed plane-induced homography provides relative depth comparisons between line segments (useful for ridge selection), the absolute depth values exhibit noticeable inaccuracies. This restricts applications requiring precise metric measurements, though it remains effective for ranking structural proximity.

3) Performance degradation under cluttered backgrounds: When inspecting towers with dense interfering line features (e.g. overlapping vegetation or secondary structures), computational latency spiked up to 420% beyond average processing times. This stems from the combinatorial complexity in AHC clustering (Section IV-A) and outlier filtering steps.

These limitations highlight key challenges for real-world deployment, particularly in adverse weather or visually congested environments. Future work should integrate deep learning to stabilize depth estimation and adopt attention mechanisms for robust line feature selection. Addressing these issues will be critical for industrial-grade reliability.

VI. CONCLUSION

This study addresses the safety protection requirements for high-altitude operations on power transmission towers by innovatively proposing an automated fall protection lanyard mounting method based on drone vision, providing an intelligent solution to replace traditional high-risk manual operations. Through multi-view geometric analysis, the system achieves 3D spatial localization of tower edges and optimal mounting point selection, overcoming the limitations of deep learning models in spatial mapping for complex scenarios. The method innovatively integrates multi-dimensional evaluation metrics, including proximity to climbing ladders, avoidance of diagonal steel obstructions, and temporal stability, ensuring that the selected mounting points comply with the "high-hook, low-use" safety principle while maintaining operational convenience. Experimental validation demonstrates that compared to conventional methods, the proposed solution significantly improves positioning accuracy and system intelligence, showing promising potential for pilot applications.

Future research can be deepened in three dimensions: First, expanding algorithm adaptability by optimizing detection models for special structures such as non-standard towers and angled steel members. Second, integrating mechanical verification modules to enable real-time assessment of load-bearing capacity at mounting points. Third, establishing open datasets and standardized testing environments to promote

collaborative technological advancement in the industry. With policy support from State Grid Corporation of China for drone-assisted operations, this technology is expected to become a standard configuration for high-altitude power operations, providing robust safety assurance for smart grid construction. Subsequent research should focus on the integration of drone swarm coordination and digital twin technology to further enhance operational reliability in complex environments.

REFERENCES

[1] W.-T. Chang, C. J. Lin, Y.-H. Lee, and H.-J. Chen, "Development of an observational checklist for falling risk assessment of high-voltage transmission tower construction workers," *International Journal of Industrial Ergonomics*, vol. 68, pp. 73–81, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016981411830132X

[2] E. National Energy Administration, *Compilation of National Electric Power Accidents and Safety Incidents (2022)*. Beijing: China Energy Media Group Co., Ltd., 10 2023, paperback, 16mo.

[3] Y. Zhang, H. Liu, and X. Chen, "Deep learning for aerial power line inspection: A review," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 11, pp. 6789–6801, 2019.

[4] L. Wang, Q. Li, and Y. Zhao, "Pl2dm: A deep learning pipeline for power line defect detection and monitoring," *Remote Sensing*, vol. 12, no. 18, p. 2987, 2020.

[5] V. Solilo, E. Okafor, and A. Nnadi, "Image enhancement strategies for uav-based power line inspection," in *Proceedings of the IEEE International Conference on Unmanned Aircraft Systems*, 2021, pp. 1122–1131.

[6] Z. Liu and M. Zhong, "Federated learning for collaborative uav-based power line inspection," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2890–2902, 2022.

[7] X. Chen, T. Zhang, and Y. Wang, "Pti-slam: Hybrid visual-inertial odometry for uav-based power tower inspection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 175, pp. 213–229, 2021.

[8] D. Lee, T. Kim, and J. Park, "Sparse-view multi-view stereo for uav-based infrastructure inspection," *Sensors*, vol. 22, no. 10, p. 3845, 2022.

[9] R. Zhang, Y. Chen, and D. Li, "Monocular slam performance evaluation for uav-based transmission tower inspection," *Autonomous Robots*, vol. 55, no. 2, pp. 321–339, 2021.

[10] R. Varghese and S. M., "Yolov8: A novel object detection algorithm with enhanced performance and robustness," in *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, 2024, pp. 1–6.

[11] A. Garcia, F. Martinez, and R. Lopez, "Learning-based multi-view stereo for aerial inspection tasks," in *Proceedings of the European Conference on Computer Vision*, 2021, pp. 334–349.

[12] S. Fang, J. Zhou, and W. Xu, "Color space transformation and edge enhancement techniques for aerial power line detection," *Sensors*, vol. 20, no. 12, p. 3456, 2020.

[13] W. Tan, F. Huang, and J. Sun, "Progressive prioritized multi-view stereo for efficient uav-based reconstruction," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1890–1897, 2021.

[14] L. Huang, G. Cheng, and X. Lin, "Challenges in textureless region reconstruction for uav vision applications," *Journal of Field Robotics*, vol. 37, no. 6, pp. 987–1004, 2020.

[15] H. Wu, L. Yang, and G. Zhao, "Semantic-aware slam for uav-based power line inspection," *Robotics and Autonomous Systems*, vol. 123, p. 103345, 2020.

[16] Y. Zhou, C. Wu, and M. Li, "Dp-mvs: Detail-preserving multi-view stereo for large-scale infrastructure modeling," *Computer Vision and Image Understanding*, vol. 223, p. 103521, 2022.

[17] Q. Zhao, M. Sun, and Y. Tang, "Outlier rejection in uav vision for power line inspection," in *Proceedings of the IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2021, pp. 123–130.

[18] J. Kim, H. Park, and S. Lee, "Gc-mvsnet++: Geometrically consistent multi-view stereo with enhanced training efficiency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4321–4335, 2022.

[19] J. Wang, Y. Liu, and H. Xu, "Multi-view stereo reconstruction for complex power infrastructure," *IEEE Access*, vol. 10, pp. 123 456–123 468, 2022.

[20] R. Yang, B. Liu, and K. Zhao, "Variable-baseline stereo vision for uav-based infrastructure inspection," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021, pp. 4567–4574.

[21] X. Li, Y. Zhang, and Z. Wang, "Pose estimation challenges in dual-uav stereo vision systems," *Journal of Intelligent and Robotic Systems*, vol. 99, no. 3, pp. 789–804, 2020.

[22] L. Zeinik-Manor and M. Irani, "Multiview constraints on homographies," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 214–223, Feb. 2002.

[23] S. A. of China (SAC), "Personal fall protection equipment–lanyard," National Standard of China, GB 24543-2009, 2009.

[24] F. Fan, Y. Jiang, K. Han, H. Li, X. Fang, and Y. Fan, "A line segment detector based on subanchor point and line segment combination method," in *2022 37th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2022, pp. 896–901.

[25] I. C. Baykal and I. C. Yilmaz, "An extremely fast pattern based line detector," in *2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2017, pp. 369–376.

[26] C. Akinlar and C. Topal, "Edlines: Real-time line segment detection by edge drawing (ed)," in *2011 18th IEEE International Conference on Image Processing*, 2011, pp. 2837–2840.

[27] I. Suárez, J. M. Buenaposada, and L. Baumela, "ELSED: Enhanced line SEgment drawing," *Pattern Recognition*, vol. 127, p. 108619, Jul. 2022.

[28] L. Zhang and R. Koch, "An efficient and robust line segment matching approach based on lbd descriptor and pairwise geometric consistency," *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 794–805, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1047320313000874

[29] M.-W. Cao, L. Li, W.-J. Xie, W. Jia, Z.-H. Lv, L.-P. Zheng, and X.-P. Liu, "Parallel k nearest neighbor matching for 3d reconstruction," *IEEE Access*, vol. 7, pp. 55 248–55 260, 2019.

[30] R. Dhivya and N. Shanmugapriya, "An efficient dbscan with enhanced agglomerative clustering algorithm," in *2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 2023, pp. 1322–1327.

APPENDIX

A. Proof of (2)

Proof: Let $\mathbf{x}_\mathrm{W} \in \mathbb{R}^3$ be an arbitrary point on the a plane $(\mathbf{n}, d)$ in the world frame. Then the $\mathbf{x}_\mathrm{W}$ satisfy,

$$\mathbf{n}^T \mathbf{x}_\mathrm{W} + d = 0 \tag{13}$$

By projecting the $\mathbf{x}_\mathrm{W}$ onto the image planes of camera at t and $t+1$, it satisfy that

$$\mathbf{x} = \mathbf{P}\mathbf{x}_\mathrm{W} \tag{14a}$$
$$\mathbf{x}' = \mathbf{P}\mathbf{R}\left(\mathbf{x}_\mathrm{W} - \mathbf{c}\right) \tag{14b}$$

The (13) can be rewritten as $-\frac{1}{d}\mathbf{n}^T\mathbf{x}_\mathrm{W} = 1$ and substituted into (14b) to obtain,

$$\mathbf{x}' = \mathbf{P}\mathbf{R}\left(\mathbf{x}_\mathrm{W} + \frac{1}{d}\mathbf{c}\mathbf{n}^T\mathbf{x}_\mathrm{W}\right) = \mathbf{P}\mathbf{R}\left(\mathbf{I} + \frac{1}{d}\mathbf{c}\mathbf{n}^T\right)\mathbf{x}_\mathrm{W} \tag{15}$$

Similarly, based on (14a), the $\mathbf{x}_\mathrm{W}$ can be rewrite as $\mathbf{x}_\mathrm{W} = \mathbf{P}^{-1}\mathbf{x}$. By putting this into (15), we have

$$\mathbf{x}' = \mathbf{PR}\left(\mathbf{I} + \frac{1}{d}\mathbf{c}\mathbf{n}^T\right)\mathbf{P}^{-1}\mathbf{x} \tag{16}$$

■

A New Hybrid Approach Based on Discrete Wavelet Transform and Deep Learning for Traffic Sign Recognition in Autonomous Vehicles

Rim Trabelsi[1], Khaled Nouri[2]
Advanced Systems Laboratory-Tunisia Polytechnic School,
University of Carthage, 2078 La Marsa, Tunis, Tunisia[1]
Tunisia Polytechnic School, University of Carthage, 2078 La Marsa, Tunis, Tunisia[2]

***Abstract*—The rapid advancement of autonomous vehicles has led to the widespread integration of advanced driver assistance systems, significantly improving vehicle control, safety, and compliance with traffic regulations. A crucial aspect of these systems is the reliable detection and recognition of traffic signs, which play a key role in managing urban traffic flow and ensuring road safety. However, traffic sign recognition remains a challenging task due to varying lighting conditions, occlusions, and diverse sign appearances. This paper presents a novel hybrid approach for efficient traffic sign recognition tailored to the needs of autonomous driving. The proposed method combines the Discrete Wavelet Transform for robust feature extraction with the powerful classification capabilities of Convolutional Neural Networks within a Deep Learning framework. The DWT effectively captures essential image characteristics while reducing noise and irrelevant details, providing a compact yet informative feature set for the CNN classifier. Extensive experiments were conducted to evaluate the performance of the system in real-world conditions. The proposed approach achieved an impressive recognition precision of 98%, demonstrating its ability to interpret and respond to traffic signs with high reliability. The results confirm the method's robustness, real-time efficiency, and suitability for deployment in intelligent transportation systems and autonomous vehicles. Overall, this study highlights the complementary strengths of DWT and CNN within the broader context of Deep Learning, offering a significant improvement over conventional traffic sign recognition techniques. The proposed system represents a promising step toward enhancing the perception capabilities of autonomous vehicles, contributing to safer and more reliable navigation in complex traffic environments.**

***Keywords*—Safety; discrete wavelet transform; traffic sign recognition; autonomous vehicles; deep learning**

I. Introduction

Scene understanding is a fundamental challenge in computer vision, encompassing critical tasks such as object detection, classification, and semantic segmentation across diverse environments [1]. Recent advances in deep learning have significantly improved performance in these areas, enabling breakthroughs in autonomous driving and robotic vision [2]. Among these tasks, Traffic Sign Recognition (TSR) has attracted particular attention because of its direct impact on road safety and navigation efficiency. Accurate and real-time TSR is essential for both human drivers and autonomous systems, especially in unfamiliar or complex driving scenarios [3]. Reliable TSR ensures that vehicles can interpret and respond appropriately to traffic regulations, which is a cornerstone for the safe deployment of intelligent transportation systems.

Despite considerable progress, TSR remains a challenging problem due to both environmental and technical factors. Environmental conditions such as varying illumination, adverse weather (e.g. rain, fog, or low-light scenarios), and occlusions often reduce detection reliability. Furthermore, traffic signs exhibit significant variability in appearance, including differences in scale, shape, and color across countries, which complicates the recognition task. On the technical side, many existing approaches suffer from limitations in computational efficiency or fail to generalize well to real-world conditions. Although intelligent vision-based systems have attempted to address these issues using advanced image processing techniques and multi-sensor fusion [4], scalability and real-time performance continue to be open challenges. These limitations highlight the need for hybrid methods that combine robustness, efficiency, and adaptability.

To address these challenges, this study proposes a novel hybrid model that integrates the Discrete Wavelet Transform (DWT) with a Convolutional Neural Network (CNN) for traffic sign recognition. DWT is used as a preprocessing step to decompose traffic sign images into multiple frequency bands, thereby preserving critical structural features while suppressing noise. CNNs, on the other hand, excel at hierarchical feature learning and robust classification. By combining these two techniques, the proposed approach leverages their complementary strengths: DWT enhances the quality of input features, while the CNN ensures accurate and efficient classification. This synergy enables improved recognition performance, particularly under adverse conditions, while maintaining real-time processing capability.

The contributions of this work are as follows: First, a DWT-based preprocessing framework is proposed to extract relevant features by capturing both low and high-frequency information, which enhances contour and detail representation while effectively reducing noise, thereby improving robustness under adverse conditions such as rain or low-light environments; second, an optimized CNN architecture is designed specifically for traffic sign recognition, striking a balance between accuracy and computational efficiency; finally, comprehensive evaluations demonstrate that the proposed approach outperforms state-of-the-art methods, particularly in challenging and adverse environments.

The remainder of this paper is structured as follows: Section II reviews related work and key challenges in TSR; Sec-

tion III details the proposed DWT-CNN methodology; Section IV presents the experimental setup, results, and comparisons; and Section V concludes with a discussion of limitations and future research directions.

II. Related Work

Recent advances in Traffic Sign Recognition (TSR) have focused on improving detection accuracy, robustness under adverse conditions, and computational efficiency. Deep learning-based approaches, particularly those leveraging convolutional neural networks (CNNs) and attention mechanisms, have dominated the field, while hybrid methods integrating traditional image processing techniques have gained traction for enhanced feature extraction. Despite these advances, most methods still face challenges in balancing accuracy, robustness, and real-time performance, especially under adverse visual conditions or on embedded platforms.

A. Deep Learning: Based Approaches

CNN-based architectures remain the cornerstone of modern TSR systems due to their hierarchical feature learning capabilities. Early successes such as Faster R-CNN and SSD laid the groundwork for robust object detection pipelines. Modified Faster R-CNN models incorporating Feature Pyramid Networks (FPN) improve small and distant sign detection [5], but they remain computationally intensive, limiting their use in real-time embedded systems. SSD variants offer real-time detection with reasonable accuracy, yet they can struggle with very small or occluded signs. YOLO models, including Sign-YOLO (attention-based YOLOv7) [6] and YOLOv8 [7], achieve an excellent speed-accuracy trade-off, yet attention-based variants often increase model complexity and may require additional tuning for low-light or motion-blurred scenarios.

Transformers, such as Vision Transformers (ViTs) [8] and Swin Transformers [9], capture long-range dependencies and complex scene context effectively, improving recognition in crowded or cluttered scenes. However, their high computational cost and large memory footprint hinder deployment on low-power devices, motivating lightweight or hybrid variants.

Overall, deep learning approaches are powerful but often trade-off robustness, speed, and hardware efficiency, motivating research into methods like DWT-CNN that aim to preserve structural features while remaining efficient.

B. Hybrid and Preprocessing-Enhanced Methods

To mitigate issues such as illumination variations, noise, and motion blur, hybrid methods combining deep learning with traditional image processing have been explored. Discrete Wavelet Transform (DWT) preprocessing enhances feature robustness by decomposing images into multi-resolution subbands [10], preserving structural details while filtering noise. Studies [11] show that wavelet-based denoising improves detection in low-visibility conditions, though the additional preprocessing step may slightly increase inference time.

Multi-scale methods like SADANet [12] improve detection across varying sign sizes but can add network complexity. Multi-sensor fusion approaches, combining LiDAR and camera data [13], [14], reduce false positives in adverse weather, yet introduce high computational requirements and challenges in real-time deployment.

In comparison, the proposed DWT-CNN integrates wavelet preprocessing directly into the CNN pipeline, achieving robustness under adverse conditions while maintaining real-time performance. This positions the proposed approach as a balanced solution that addresses key limitations of both pure deep learning and hybrid methods, providing a strong foundation for real-world traffic sign recognition systems.

III. Proposed Method

In this work, this study proposes a novel hybrid architecture that integrates DWT with a CNN for robust and efficient Traffic Sign Recognition (TSR). The core idea is to leverage the multi-resolution analysis capability of DWT to enhance feature robustness under challenging conditions such as noise, motion blur, and lighting variations, while preserving a lightweight design suitable for real-time applications.

The proposed traffic sign recognition method is based on a hybrid architecture that combines the strengths of the Discrete Wavelet Transform (DWT) and Convolutional Neural Networks (CNNs) to ensure both robustness and efficiency in real-world scenarios. The overall pipeline is designed to enhance feature extraction under challenging conditions such as noise, motion blur, and varying illumination, while maintaining the low computational cost required for real-time deployment. The system begins by preparing and preprocessing a diverse set of traffic sign images to ensure uniform input and reduce variability, where input images are standardized to 128×128 pixels, and converted to grayscale to optimize computational efficiency. Next, the images are passed through a DWT-based transformation to extract multi-resolution frequency components, allowing the model to focus on structural details and suppress irrelevant noise. These enhanced representations are then fed into a lightweight CNN specifically designed for traffic sign classification. The model is trained with optimized strategies to prevent overfitting and maximize generalization. Finally, extensive evaluations are conducted to assess the system's accuracy, speed, and robustness under various environmental conditions. A comprehensive overview of the model is visually presented in Fig. 1 and the following subsections describe each phase of the proposed method in detail.

1) Preprocessing pipeline: Traffic signs suffer two main degradation types: natural and human-induced. Natural degradation is typically caused by long-term exposure to ultraviolet (UV) radiation and the use of retro-reflective materials, which may alter the sign's color and reduce its visibility. On the other hand, human-related degradation can take many forms, affecting the shape, color, components, or even the entire structure of the sign. To ensure accurate recognition of such signs despite potential deterioration, robust image preprocessing is essential. In the initial stage, the input images are standardized to improve compatibility with the Convolutional Neural Network (CNN). This includes resizing all images to a fixed dimension (640×640 pixels), then converting them to grayscale to reduce computational complexity while retaining edge and structural information.

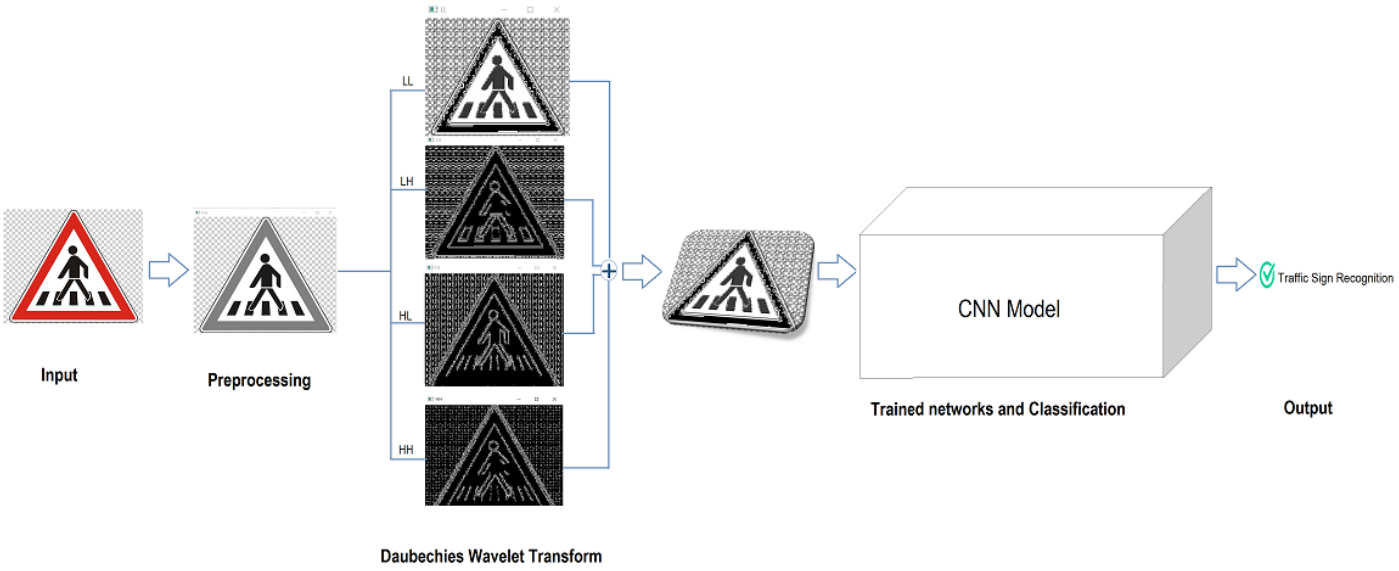


Fig. 1. Proposed TSR system based on wavelet transform and CNN: (a) Preprocessing, (b) Daubechies DWT: Decomposition Sub-bands, (c) Feature Fusion, and (d) CNN model.

2) DWT-Based feature enhancement: The DWT addresses key challenges in traffic sign recognition such as noise robustness and multi-scale feature extraction by decomposing images into frequency sub-bands. Unlike traditional methods, it preserves spatial and frequency information simultaneously, making it ideal for analyzing degraded signs [15].

There are two main types of wavelet transforms: the Continuous Wavelet Transform (CWT), designed for continuous signal decomposition, and the Discrete Wavelet Transform (DWT), commonly used in practical applications for analyzing both signals and images. The DWT decomposes input data into frequency sub-bands and is favored for its computational efficiency and simplicity.

Mathematically, the DWT is defined by Eq. 1, where $x(k)$ is a discrete-time signal with k samples and ψ is the discrete mother wavelet used for decomposition level i:

$$DWT_{ik} = \sum x(k)\psi_{i,k}(t) \tag{1}$$

Here, $\psi_{i,k}$ are scaled and translated versions of the mother wavelet $\psi(x)$:

$$\psi_{i,k}(x) = 2^{-i/2}\psi(2^{-i}x - k) \tag{2}$$

where i and k represent the dilation and translation parameters, respectively.

Various mother wavelets can be employed, such as Haar, Daubechies, Meyer, and Mexican Hat, depending on the characteristics of the application [16]. In this study, the Daubechies wavelet is employed due to its orthogonality and compact support, making it particularly effective for discrete signal processing.

To compute the DWT, the signal is passed through a low-pass filter $g[n]$ and a high-pass filter $h[n]$, as shown in Eq. 3:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k]g[n-k] \tag{3}$$

The Daubechies wavelet is adopted in this study due to its compact support and orthogonality, which efficiently capture localized signal features. The one-level decomposition (Fig. 2) partitions the image into:

- LL: Low-frequency components (approximation)
- LH/HL: (horizontal/vertical edges)
- HH: High-frequency components (diagonal details)

This decomposition process is illustrated in Fig. 2. The maximum decomposition level depends on the number of samples and the sampling frequency f_s. The frequency ranges for approximation and detail sub-bands at level i are given by:

$$A_i = \left[0, \frac{f_s}{2^{i+1}}\right], \tag{4}$$

$$D_i = \left[\frac{f_s}{2^{i+1}}, \frac{f_s}{2^1}\right]. \tag{5}$$

The 1-level decomposition was chosen to balance computational cost and feature granularity, as higher levels showed diminishing returns in preliminary experiments.

Instead of feeding raw images into the CNN, the decomposed sub-bands (LL, LH, HL, HH) are used as inputs, as illustrated in Fig. 3. This allows the network to learn more robust and localized features, particularly edges and shape patterns, which are crucial for accurate traffic sign recognition.

Using only the LL sub-band as input to a CNN may result in the loss of crucial information present in other sub-bands

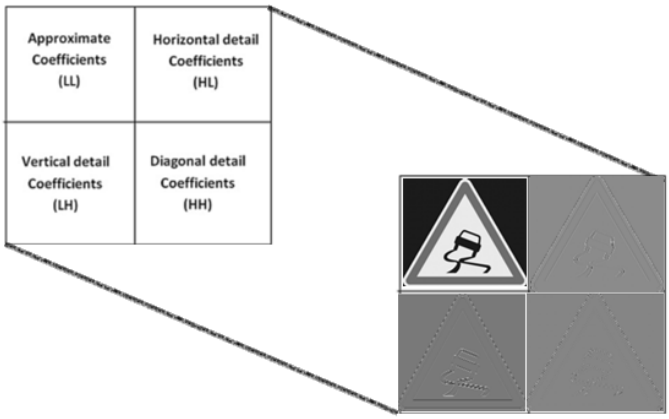


Fig. 2. Wavelet decomposition coefficients for the first level.

(LH, HL, and HH), which contain important details. By including all sub-bands as input, the model can manipulate details at various frequencies, ensuring a comprehensive representation of the image. The LH sub-band captures intermediate-frequency details, such as textures and edges, the HL sub-band contains high-frequency information, such as small details and color variations, and the HH sub-band contains very high-frequency information, such as very fine details and color variations. This approach enhances the model's ability to capture important image features, leading to improved performance and accuracy.

Fig. 3 presents the results obtained from each of the four sub-bands: LL, LH, HL, and HH.

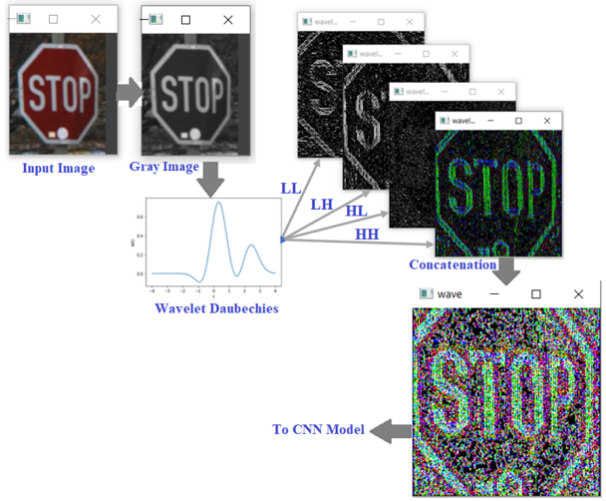


Fig. 3. Wavelet sub-band decomposition and concatenation (LL, LH, HL, HH).

In this work, a one-level 2D DWT decomposition is applied to each input image prior to classification to enhance the CNN's performance and improve feature representation.

3) CNN architecture: In this study, a DWT-based Traffic Sign Recognition System is introduced to enhance the feature extraction capability, improve detection performance, and reduce memory consumption and overall model size. As illustrated in Fig. 4, the system leverages the sub-bands generated by the Discrete Wavelet Transform (DWT). The main objective of this DWT-based approach is to extract robust and discriminative features from the DWT outputs and the backbone network, thereby improving detection accuracy under various conditions. Additionally, this mechanism guides the network's attention toward the most informative spatial regions particularly useful when traffic signs appear in cluttered or complex scenes.

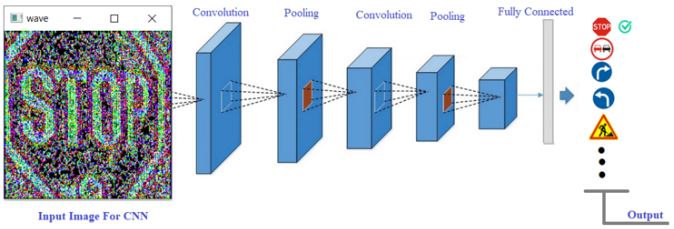


Fig. 4. Overall architecture of the proposed framework for TSR.

The convolutional layer [17] is a fundamental building block in CNNs and typically consists of three sequential components: convolution, activation, and pooling. In image processing tasks, the input is treated as a two-dimensional array, and the convolution operation applies a filter of size $S \times T$ over the input of size $M \times N$, producing an output of the same dimensions. The operation is defined as:

$$U^p_{mn} = \sum_{s=0}^{S-1} \sum_{t=0}^{T-1} h^{p-1}_{st} \cdot O^{p-1}_{m+s,n+t} + \theta^p_{mn} \tag{6}$$

Here, U^p_{mn} denotes the pre-activation value at location (m, n) in layer p, h^{p-1}_{st} is the (s, t)-th weight of the filter, $O^{p-1}_{m+s,n+t}$ is the corresponding input from the previous layer, and θ^p_{mn} is the bias term.

Subsequently, an activation function is applied. For instance, using the ReLU activation, the post-activation output becomes:

$$O^p_{mn} = \text{ReLU}(U^p_{mn}) \tag{7}$$

In the activation function layer, where no parameters are trained, the derivative for error backpropagation is calculated as follows:

$$\begin{aligned} \frac{\partial E}{\partial O^{p-1}_{mn}} &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \frac{\partial E}{\partial U^p_{mn}} \cdot \frac{\partial U^p_{mn}}{\partial O^{p-1}_{mn}} \\ &= \frac{\partial E}{\partial U^p_{mn}} \cdot \frac{\partial ReLU(O^{p-1}_{mn})}{\partial O^{p-1}_{mn}} \\ &= \begin{cases} \frac{\partial E}{\partial U^p_{mn}} & (O^{p-1}_{mn} > 0) \\ 0 & (O^{p-1}_{mn} \leq 0) \end{cases} \end{aligned} \tag{8}$$

When a pooling layer follows a convolutional layer, it aggregates spatial information within localized regions of the feature map to reduce dimensionality and retain salient

features. The general form of the pooling operation at position (m, n) in the p-th layer is defined as:

$$U_{mn}^{p} = \left(\frac{1}{S \times T} \sum_{(i,j) \in D_{mn}} (O_{ij}^{p-1})^{g} \right)^{\frac{1}{g}} \tag{9}$$

In this equation, O_{ij}^{p-1} denotes the output of the unit at position (i, j) in the $(p-1)$-th layer, and D_{mn} represents the pooling window of size $S \times T$ centered around (m, n). The parameters S and T are typically chosen to match the dimensions of the convolutional filter. The parameter g controls the type of pooling: setting $g = 1$ yields average pooling, while $g \to \infty$ approximates max pooling.

For example, setting $g = 1$ results in the average pooling operation, where the output is the arithmetic mean of the values within the pooling region:

$$U_{mn}^{p} = \frac{1}{S \times T} \sum_{(i,j) \in D_{mn}} O_{ij}^{p-1} \tag{10}$$

The averaging operation in the pooling layer reduces the spatial dimensions of the input while preserving essential features, thereby improving computational efficiency. When the pooling parameter g tends toward infinity, the operation becomes equivalent to max pooling, where the output at position (m, n) in the p-th layer is determined by selecting the maximum value within the pooling window D_{mn}. In this case, the pooling operation simplifies to:

$$U_{mn}^{p} = \max_{(i,j) \in D_{mn}} \{O_{ij}^{p-1}\} \tag{11}$$

Pooling layers, whether average or max, perform spatial downsampling to reduce feature map resolution while retaining discriminative information. Max pooling preserves the most dominant activations, while average pooling computes the mean of local regions. Since pooling layers do not involve trainable parameters, backpropagation in these layers is limited to the propagation of gradients from the current layer p to the previous layer $p-1$.

For average pooling, the gradient of the loss function E with respect to the output O_{mn}^{p-1} of the previous layer is computed by equally distributing the gradient across the pooling window. The corresponding gradient expression is given by:

$$\begin{aligned} \frac{\partial E}{\partial O_{mn}^{p-1}} &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \frac{\partial E}{\partial U_{mn}^{p}} \cdot \frac{\partial U_{mn}^{p}}{\partial O_{mn}^{p-1}} \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \frac{\partial E}{\partial U_{mn}^{p}} \cdot \left(\frac{1}{S \times T} \sum_{(i,j) \in D_{mn}} \frac{\partial O_{ij}^{p-1}}{\partial O_{mn}^{p-1}} \right) \end{aligned} \tag{12}$$

In contrast, backpropagation through max pooling involves identifying the location of the maximum value within each pooling region. The gradient is passed only to the unit that contributed the maximum activation. This can be formulated as:

$$\begin{aligned} \frac{\partial E}{\partial O_{mn}^{p-1}} &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \frac{\partial E}{\partial U_{mn}^{p}} \cdot \frac{\partial U_{mn}^{p}}{\partial O_{mn}^{p-1}} \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \frac{\partial E}{\partial U_{mn}^{p}} \cdot \frac{\partial}{\partial O_{mn}^{p-1}} \left(\max_{(i,j) \in D_{mn}} \{O_{ij}^{p-1}\} \right) \end{aligned} \tag{13}$$

In addition to pooling, normalization layers are often employed to stabilize and accelerate the training process. A common form of normalization is defined as:

$$U_{mn}^{p} = \frac{O_{mn}^{p-1} - \mu_{mn}}{\sqrt{c + \sigma_{mn}^{2}}} \tag{14}$$

where U_{mn}^{p} is the normalized output at position (m, n), O_{mn}^{p-1} is the input value from the previous layer, μ_{mn} and σ_{mn}^{2} are the local mean and variance computed over the pooling window D_{mn}, and c is a small constant added for numerical stability. These statistical quantities are computed as:

$$\mu_{mn} = \frac{1}{S \times T} \sum_{(i,j) \in D_{mn}} O_{ij}^{p-1} \tag{15}$$

$$\sigma_{mn}^{2} = \frac{1}{S \times T} \sum_{(i,j) \in D_{mn}} (O_{ij}^{p-1} - \mu_{mn})^{2} \tag{16}$$

These normalization steps help maintain consistent activation distributions across layers, which contributes to faster convergence and improved generalization during training.

4) System workflow overview: Fig. 4 illustrates the overall architecture of the proposed CNN, highlighting the sequence of convolution, activation, pooling, and normalization layers.

The pseudocode in Algorithm 1 describes the core stages of the proposed DWT-based traffic sign recognition system. It begins by converting the input image to grayscale and initializing all key components of the model, including a DWT layer, a convolutional backbone network, a neck module, and a detection head. During training, the image undergoes preprocessing and is passed through the DWT layer to extract frequency-based features. These features are then processed by the convolutional backbone, which comprises several layers of convolution, batch normalization, ReLU activations, dropout, and max pooling. The refined feature maps are passed through the neck and then to the detection head, which predicts both class probabilities and bounding box coordinates. The training loop optimizes the model using a combination of classification and localization losses. During inference, the input image follows the same processing pipeline, leading to the final prediction of bounding box coordinates and the corresponding class label.

A key originality of the proposed architecture lies in the explicit integration of the DWT as the first processing stage of the network. Unlike conventional CNN-based TSR systems that rely solely on spatial-domain features, the model leverages

Algorithm 1: Pseudocode of DWT-Based Traffic Sign Recognition System

```
Input: input_image
Output: bounding_box, class
Convert input_image to grayscale;
Initialize model components:;
    Define DWT layer (discrete wavelet transform);
    Define CNN backbone with convolutional, batchnorm, ReLU, dropout, max-pooling layers;
    Define Neck: convolutional layer with ReLU;
    Define Head: fully-connected layers for classification and bounding box regression;
Define data augmentation pipeline;
Load training dataset from CSV and image directory;
Initialize optimizer and loss functions (CrossEntropy and MSE);
while training not converged do
    Read batch of images, labels, and bounding boxes;
    Move data to device (GPU or CPU);
    Zero gradients;
    Forward pass:;
        Apply DWT to input images;
        Pass through CNN backbone:;
            - Conv + BatchNorm + ReLU + Dropout (64 filters);
            - Conv + BatchNorm + ReLU + Dropout (128 filters);
            - MaxPooling;
            - Conv + BatchNorm + ReLU + Dropout (256 filters);
            - MaxPooling;
        Pass through Neck (Conv + ReLU);
        Pass through Detection Head (FC for class and bbox);
    Compute classification and bounding box loss;
    Total loss = classification loss + bbox loss;
    Backpropagate loss and update model weights;
    Log average loss per epoch;
Save trained model weights to file;
Plot loss curve over epochs;
while testing do
    Read input image;
    Convert image to grayscale;
    Forward pass:;
        Apply DWT to input image;
        Pass through CNN backbone;
        Pass through Neck and Detection Head;
    Apply non-maximum suppression to filter redundant detections;
```

frequency-domain information to enhance robustness against noise, illumination changes, and motion blur. This design allows the backbone to operate on enriched multi-resolution representations, reducing the loss of fine structural details typically discarded in early convolutional layers. Moreover, by embedding the wavelet decomposition directly into the training pipeline rather than using it as a separate preprocessing step, the system jointly optimizes spatial and frequency features, ensuring better generalization under adverse conditions. This tight coupling of DWT with CNN layers distinguishes the proposed approach from existing hybrid methods and constitutes the main source of its improved performance and resilience.

IV. Results and Evaluation

This section demonstrates the effectiveness of the proposed DWT-CNN framework. First, the datasets used for training and evaluation are presented. Next, the implementation setup and evaluation protocol are detailed. Finally, a comparative analysis between the proposed method and other state-of-the-art approaches is provided.

1) Traffic signs dataset: The German Traffic Sign Recognition Benchmark (GTSRB) is a popular dataset for traffic sign recognition research [18] due to its large size (over 50,000 images of 43 different classes of traffic signs), diverse classes shown in Fig. 5 (including speed limits, priority, prohibitory, and warning signs), high-quality annotations, and wide use as a benchmark. Additionally, the dataset is publicly available and free to use, making it easily accessible for researchers. These factors have contributed to GTSRB's popularity as a benchmark for evaluating traffic sign recognition algorithms in recent years.

2) Implementation details: The proposed DWT-CNN model was implemented and evaluated using the German Traffic Sign Recognition Benchmark (GTSRB). For this study, a selected subset of GTSRB subclasses particularly relevant to real-world traffic scenarios was used. Each input image was resized to a resolution of 640×640 pixels for both the training and testing phases.

The experiments were conducted on a high-performance machine equipped with an Intel 13th Gen Core(TM) i7-13650HX CPU clocked at 2.60 GHz, 32 GB of RAM, and an NVIDIA GeForce RTX 4060 GPU with 8 GB of memory. The implementation was developed with CUDA version 12.7 and cuDNN version 9.1.0 for GPU acceleration.

This hardware configuration ensured efficient training and inference, especially for high-resolution input images and large-scale datasets.

The model was implemented using Python and PyTorch. The training process incorporated standard data augmentation techniques, including random rotation, horizontal flipping, and color jittering, to enhance generalization. Optimization was carried out using the Adam optimizer with an empirically tuned learning rate. The loss function combined cross-entropy for classification with mean squared error for bounding box regression.

3) Evaluation metrics: To evaluate the accuracy of the proposed method, this study employs standard metrics widely used in object detection and classification, including precision, recall, accuracy, and mean average precision (mAP). These metrics provide comprehensive insights into the method's performance in terms of correctness, completeness, and effectiveness, and also allow for fair comparisons with existing approaches in the literature.

Precision measures the proportion of correctly predicted positive samples out of all positive predictions, reflecting the model's accuracy in identifying relevant objects. Formally, it is defined as follows [19]:



Fig. 5. Summary of the dataset classes.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{17}$$

where:

- TP (True Positive): Correctly predicted object,
- FP (False Positive): Incorrectly predicted object.

Recall, also known as *sensitivity* or *true positive rate*, measures the ability of the model to detect all relevant instances of a class. It is computed as [20]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{18}$$

where FN (False Negative) refers to instances where the model failed to detect existing objects.

Accuracy assesses the overall correctness of the model's predictions, considering both positive and negative classifications:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{19}$$

where TN (True Negative) represents the correct prediction of the absence of objects.

Finally, mean average precision (mAP) is a key performance indicator in object detection tasks. It is computed by averaging the average precision (AveP) across all object classes:

$$\text{mAP} = \frac{1}{n} \sum_{k=1}^{n} \text{AveP}_k \tag{20}$$

where AveP_k is the average precision for class k, and n is the total number of classes.

In summary, relying on these validation metrics is fundamental to rigorously assessing the reliability of the proposed method. They ensure that the evaluation is not biased toward a single criterion but instead provides a holistic view of detection performance, which is essential for benchmarking against other state-of-the-art models.

4) Result and Analysis: As evidenced by the confusion matrix in Fig. 6, the model achieves high classification accuracy across diverse sign categories, with minimal misclassifications even for visually similar classes. The synergy of DWT's noise resilience and the CNN's learning capability ensures reliable performance under real-world challenges such as occlusions and weather distortions.

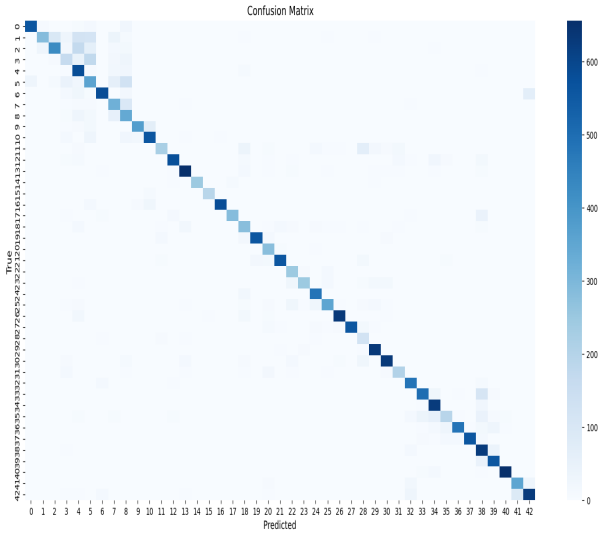


Fig. 6. Confusion matrix showing the performance of the proposed classification model.

Rigorous validation is essential in traffic sign recognition, where errors can directly affect road safety. Metrics such as precision, recall, F1-score, and mean average precision (mAP) not only quantify accuracy but also reveal the robustness of the model under varying conditions. High precision reduces false positives, which is critical to avoid incorrect driving actions, while high recall minimizes false negatives, ensuring that no critical sign is overlooked. These complementary measures together validate both the correctness and completeness of the proposed DWT-CNN model.

To assess the efficacy of the DWT-CNN model, this study conducts experiments on the GTSRB dataset, which comprises 43 subclasses grouped into four superclasses: prohibitory, mandatory, danger, and other. As shown in Table I, the model achieves strong performance across these categories, with par-

ticularly high accuracy for critical prohibitory and mandatory signs (e.g. stop: F1-score 0.96, speed_limit_20: 0.98). The results demonstrate the effectiveness of combining DWT's noise-robust feature extraction with CNN's discriminative learning, particularly in challenging real-world conditions.

TABLE I. VALIDATION OF THE PROPOSED METHOD ON SUB-CLASSES OF THE GTSRB

Class	Precision	Recall	F1-Score
0: speed_limit_20	0.98	0.97	0.98
1: speed_limit_30	0.86	0.90	0.88
2: speed_limit_50	0.78	0.97	0.86
3: speed_limit_60	0.97	0.97	0.97
4: speed_limit_70	0.98	0.96	0.97
5: speed_limit_80	0.93	0.87	0.90
6: restriction_ends_80	0.95	0.93	0.94
7: speed_limit_100	0.89	0.81	0.85
8: speed_limit_120	0.96	0.76	0.85
9: no_overtaking	0.91	0.78	0.84
10: no_overtaking_trucks	0.83	0.85	0.84
11: priority_at_next	0.83	0.72	0.77
12: priority_road	0.95	0.84	0.89
13: give_way	0.91	0.91	0.91
14: stop	1.00	0.92	0.96
15: no_traffic_both_ways	0.97	0.91	0.94
16: no_trucks	0.97	0.90	0.93
17: no_entry	0.96	0.79	0.87
18: danger	0.83	0.72	0.77
19: bend_left	0.98	0.93	0.95
20: bend_right	0.98	0.81	0.89
21: bend_double	0.95	0.94	0.95
22: bumpy road	0.97	0.96	0.97
23: slippery	0.97	0.90	0.93
24: road_narrows	0.88	0.84	0.86
25: construction	0.87	0.74	0.80
26: traffic_signal	1.00	0.99	0.99
27: pedestrian_crossing	1.00	0.92	0.96
28: children_crossing	0.96	0.86	0.91
29: bicycles	0.98	0.97	0.98
30: snow	0.99	0.98	0.99
31: animals	0.97	0.96	0.96
32: restriction_ends	0.97	0.87	0.92
33: go_right	0.95	0.98	0.96
34: go_left	0.99	0.97	0.98
35: go_straight	0.97	0.88	0.92
36: go_straight_right	0.98	0.97	0.98
37: go_straight_left	1.00	0.93	0.96
38: keep_right	0.97	0.94	0.96
39: keep_left	0.97	0.88	0.92
40: roundabout	1.00	0.92	0.96
41: restriction_ends_overtaking	0.98	0.96	0.97
42: restriction_ends_overtaking_trucks	0.96	0.91	0.93

Table II compares the proposed DWT-CNN model to state-of-the-art detectors based on key performance metrics: mean Average Precision (mAP), inference speed (FPS), memory usage (MB), and model size (number of parameters in millions). The DWT-CNN achieves a mAP of 0.97, outperforming all baseline models, including Faster R-CNN variants (ranging from 0.9062 to 0.9508) and YOLOv7/YOLOv8 (0.92–0.93), while maintaining real-time performance with 88.83 FPS. This is significantly faster than two-stage detectors such as Faster R-CNN (8.11–17.08 FPS) and even lightweight models like SSD MobileNet (66.03 FPS). Furthermore, the model accomplishes this with only 3.6 million parameters—substantially fewer than YOLOv7 (70.31M) and ResNet-based architectures (12.89–62.38M). Although the memory footprint (4493.43 MB) is higher than that of SSD MobileNet (94.70 MB), the trade-off is justified by a substantial gain in accuracy (+35.36% mAP) and speed (+22.8 FPS). These results highlight DWT-CNN's strong balance between accuracy and efficiency, making it well-suited for real-time traffic sign recognition tasks where high precision is essential.

Visual results in Fig. 7 illustrate the model's ability to accurately detect and classify traffic signs under challenging conditions from the GTSRB dataset. The test images include difficult scenarios such as low resolution, motion blur, occlusion, and poor lighting. Despite these adverse conditions, the proposed DWT-CNN model maintains high prediction accuracy, demonstrating both its robustness and strong generalization capabilities.

Table III presents a comparative analysis of the average precision and recall obtained by the proposed DWT-CNN model and several state-of-the-art traffic sign detection (TSD) approaches. The method achieves a precision of 0.98 and a recall of 0.97, matching YOLOv8 in precision while outperforming it by 3 percentage points in recall ($0.94 \rightarrow 0.97$). The DWT-CNN also surpasses two-stage detectors such as Faster R-CNN by 8% in precision and 10% in recall, and SADANet by 16% on both metrics. Compared to other single-stage models like YOLOv7 and YOLOv5, the model maintains highly competitive precision (e.g. YOLOv7: 0.97) and offers a substantial improvement in recall (YOLOv5: +14%). These results demonstrate the effectiveness of integrating Discrete Wavelet Transform (DWT) for noise-robust feature extraction with CNN-based classification, leading to a significant reduction in false negatives,a critical factor for traffic sign detection in safety-critical applications such as autonomous driving.

V. CONCLUSION

In this work, this study proposed a novel DWT-CNN architecture for traffic sign recognition, combining the robustness of Discrete Wavelet Transform-based feature extraction with the discriminative power of convolutional neural networks. Extensive experiments on the GTSRB dataset demonstrated that the model outperforms several state-of-the-art detectors, achieving high precision (98%) and recall (97%) while maintaining real-time inference speeds (88 FPS). The DWT-CNN showed strong resilience under adverse visual conditions such as blur, occlusion, and low lighting, confirming its robustness and generalization capability. These results highlight the model's suitability for real-world applications, particularly in safety-critical environments like autonomous driving systems. Future work will focus on deploying the model on embedded platforms and improving its performance in scenarios involving high-speed moving targets. Beyond these perspectives, several open challenges remain. For instance, how the DWT-CNN would perform when integrated into large-scale multimodal perception systems, or how it could adapt to unseen traffic sign classes across different countries, are questions that deserve further investigation. Addressing such issues could inspire new research directions and strengthen the impact of wavelet-based deep learning in intelligent transportation systems.

ACKNOWLEDGMENT

The authors would like to dedicate this research to their families for their continuous support and understanding. In particular, this work is dedicated to the memory of my father, Mr. Khalifa Trabelsi, in deep gratitude for his unwavering love, guidance, and inspiration.

TABLE II. PERFORMANCE COMPARISON OF THE PROPOSED DWT-CNN MODEL WITH STATE-OF-THE-ART OBJECT DETECTORS ACROSS KEY EVALUATION METRICS

Model	mAP (%)	FPS	Memory (MB)	Parameters (10^6)
Faster R-CNN ResNet 101 [21]	95.08	8.11	6134.71	62.38
Faster R-CNN ResNet 50 [21]	91.52	9.61	5226.45	43.34
Faster R-CNN Inception V2 [21]	90.62	17.08	2175.21	12.89
SSD MobileNet [21]	61.64	66.03	94.70	94.70
Zang et al. [22]	93.36	62	–	–
YOLOv7 [23]	93.00	35	321.67	70.31
YOLOv8 [24]	92.00	50	321.67	60
DWT-CNN (Ours)	0.97	88.83	4493.43	3,6

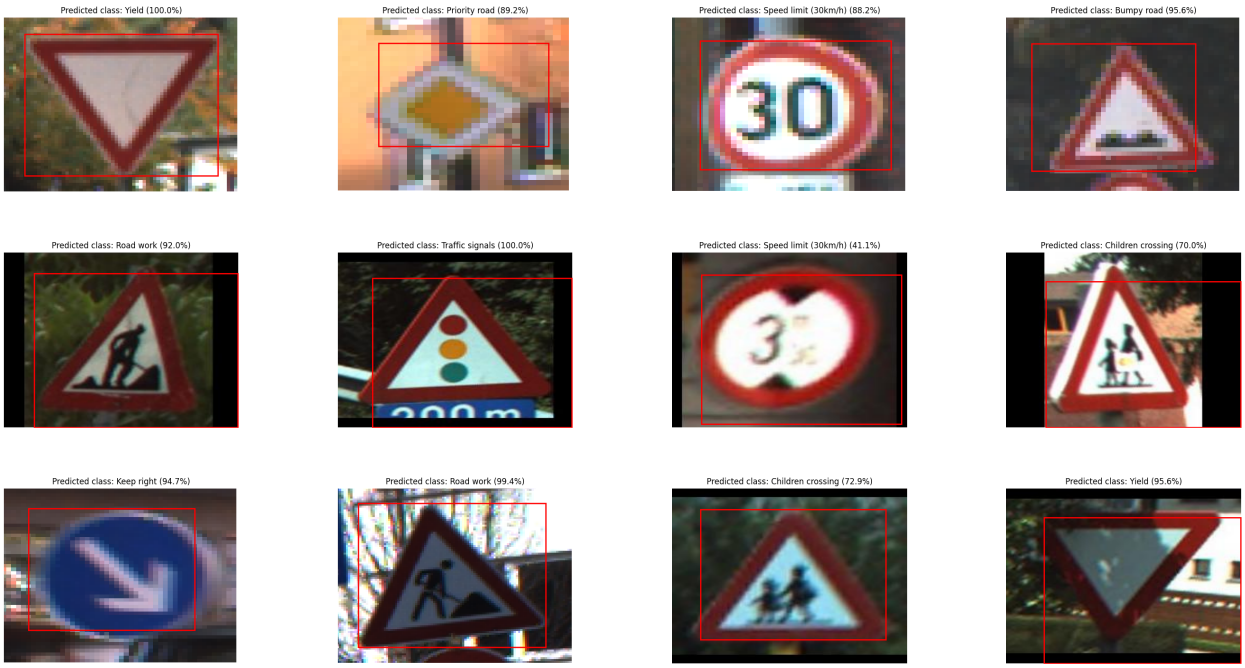


Fig. 7. Predicted class and confidence score for each image under various conditions using the proposed DWT-CNN model.

TABLE III. COMPARISON OF AVERAGE PRECISION AND RECALL OF THE PROPOSED METHOD WITH OTHER TSD APPROACHES

Model	Precision	Recall
SADANet [25]	0.82	0.81
Faster R-CNN [26]	0.90	0.87
YOLOv7 [23]	0.97	0.96
YOLOv8 [24]	0.98	0.94
Yolov5 [27]	0.85	0.83
DWT-CNN (Ours)	0.98	0.97

REFERENCES

[1] XIE, Guohuan, HESHAM, Syed Ariff Syed, GUO, Wenya, et al. A comprehensive survey on video scene parsing: advances, challenges, and prospects. [en ligne] arXiv preprint arXiv:2506.13552, 2025.

[2] SAH, Chandan Kumar, SHAW, Ankit Kumar, LIAN, Xiaoli, et al. Advancing autonomous vehicle intelligence: deep learning and multimodal LLM for traffic sign recognition and robust lane detection. [en ligne] arXiv preprint arXiv:2503.06313, 2025.

[3] HUYNH, Kha Tu, LE, Thi Phuong Linh, TRAN, Thien Khai, et al. A deep learning model of traffic signs in panoramic images detection. Intelligent Automation & Soft Computing, vol. 37, no. 1, 2023.

[4] MANI, Murali Krishnan, RAJAGOPAL, Sonaa, KAVITHA, D., et al. Deep learning-based traffic sign detection and recognition for autonomous vehicles. Digital Twin and Blockchain for Smart Cities, 2024, p. 407-428.

[5] SURESHA, R., MANOHAR, N., KUMAR, G. Ajay, et al. Recent advancement in small traffic sign detection: approaches and dataset. IEEE Access, 2024.

[6] MAHADSHETTI, Ruturaj, KIM, Jinsul, UM, Tai-Won. Sign-YOLO: traffic sign detection using attention-based YOLOv7. IEEE Access, vol. 12, p. 132689-132700, 2024.

[7] SOYLU, Emel, SOYLU, Tuncay. A performance comparison of YOLOv8 models for traffic sign detection in the Robotaxi-full scale autonomous vehicle competition. Multimedia Tools and Applications, vol. 83, no. 8, p. 25005-25035, 2024.

[8] DOSOVITSKIY, Alexey, BEYER, Lucas, KOLESNIKOV, Alexander, et al. An image is worth 16x16 words: transformers for image recognition at scale. [en ligne] arXiv preprint arXiv:2010.11929, 2020.

[9] LIU, Ze, LIN, Yutong, CAO, Yue, et al. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, p. 10012-10022.

[10] TRABELSI, Rim, NOURI, Khaled, AMMARI, Imen. Enhancing traffic sign recognition through Daubechies discrete wavelet transform and convolutional neural networks. In: 2023 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET), 2023, p. 1-6.

[11] LIU, Zhanwen, SHEN, Chao, QI, Mingyuan, et al. SADANet: integrating scale-aware and domain adaptive for traffic sign detection. IEEE Access, vol. 8, p. 77920-77933, 2020.

[12] SURESHA, R., MANOHAR, N., KUMAR, G. Ajay, et al. Recent advancement in small traffic sign detection: approaches and dataset. IEEE Access, 2024.

[13] CHEN, Peng, ZHAO, Xinyu, ZENG, Lina, et al. A review of research on SLAM technology based on the fusion of LiDAR and vision. Sensors, vol. 25, no. 5, p. 1447, 2025.

[14] WANG, Zhangjing, WU, Yu, NIU, Qingqing. Multi-sensor fusion in automated driving: a survey. IEEE Access, vol. 8, p. 2847-2868, 2019.

[15] TRABELSI, Rim, NOURI, Khaled. Boosting facial recognition accuracy through Daubechies wavelet techniques and deep convolutional models. In: 2024 International Symposium of Systems, Advanced Technologies and Knowledge (ISSATK), 2024, p. 1-6.

[16] GALAN-URIBE, Ervin, AMEZQUITA-SANCHEZ, Juan P., MORALES-VELAZQUEZ, Luis. Supervised machine-learning methodology for industrial robot positional health using artificial neural networks, discrete wavelet transform, and nonlinear indicators. Sensors, vol. 23, no. 6, p. 3213, 2023.

[17] YAGAWA, Genki, OISHI, Atsuya. Computational mechanics with deep learning: an introduction. Springer Nature, 2022.

[18] Belgium Traffic Sign Dataset. [en ligne] Disponible sur: https://btsd.ethz.ch/shareddata et https://www.kaggle.com/datasets/gtsrb-german-traffic-sign

[19] ZHU, Wen, ZENG, Nancy, WANG, Ning, et al. Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations. NESUG Proceedings: Health Care and Life Sciences, vol. 19, p. 67, 2010.

[20] POWERS, David. Ailab. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness & correlation. J. Mach. Learn. Technol., vol. 2, no. 22293981, p. 01, 2011.

[21] ARCOS-GARCÍA, Álvaro, ALVAREZ-GARCIA, Juan A., SORIA-MORILLO, Luis M. Evaluation of deep neural networks for traffic sign detection systems. Neurocomputing, vol. 316, p. 332-344, 2018.

[22] ZHANG, Jianming, LV, Yaru, TAO, Jiajun, et al. A robust real-time anchor-free traffic sign detector with one-level feature. IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 8, no. 2, p. 1437-1451, 2024.

[23] WANG, Chien-Yao, BOCHKOVSKIY, Alexey, LIAO, Hong-Yuan Mark. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, p. 7464-7475.

[24] FARHAT, Wajdi, RHAIEM, Olfa Ben, FAIEDH, Hassene, et al. YOLO-TSR: a novel YOLOv8-based network for robust traffic sign recognition. Transportation Research Record, 2025.

[25] LIU, Zhanwen, SHEN, Chao, QI, Mingyuan, et al. SADANet: integrating scale-aware and domain adaptive for traffic sign detection. IEEE Access, vol. 8, p. 77920-77933, 2020.

[26] SHAO, Faming, WANG, Xinqing, MENG, Fanjie, et al. Improved faster R-CNN traffic sign detection based on a second region of interest and highly possible regions proposal network. Sensors, vol. 19, no. 10, p. 2288, 2019.

[27] ORTATAŞ, Fatma Nur, KAYA, Mahir. Performance evaluation of YOLOv5, YOLOv7, and YOLOv8 models in traffic sign detection. In: 2023 8th International Conference on Computer Science and Engineering (UBMK), 2023, p. 151-156.

A Computer-Aided Diagnosis System for Ulcerative Colitis Classification Using Vision Transformer

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]
Department of Artificial Intelligence and Machine Learning, Manipal University Jaipur, Jaipur, Rajasthan, India[2,3]
Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India[4]

***Abstract*—An unhealthy digestive condition that inflames the colon is called ulcerative colitis (UC). Utilising colonoscopy information to assess disease severity is a laborious process that concentrates on the most severe anomalies. The severity of this condition can significantly impact a patient's quality of life. Current diagnostic methods, primarily colonoscopy, for assessing UC severity are subjective and prone to inter-observer variability, hindering accurate staging and personalized treatment. Colonoscopies are currently used by doctors to diagnose the severity of ulcerative colitis, yet this might be imprecise due to physician variance. As such, to deliver optimal outcomes, automated and precise technology is required. The current study introduces UC-visionNet, an automated approach that classifies ulcerative colitis severity based on colonoscopy image analysis using vision transfer techniques. UC-visionNet makes use of vision transformers, which are pre-trained deep learning models that have shown to be quite successful in image analysis applications. To classify ulcerative colitis severity, these models are "fine-tuned" using the LIMUC (Labeled Images for Ulcerative Colitis) dataset. Compared to conventional colonoscopy procedures, using UC-visionNet for image analysis may be faster, enhancing patient satisfaction and increasing healthcare effectiveness. In contrast to state-of-the-art techniques, the suggested model performs quantitatively better on the LIMUS dataset. After using Vision transformer (ViT) on the LIMUC dataset, the current study attained a 96% training accuracy. UC-visionNet offers a promising automated solution for accurate and efficient UC severity classification.**

***Keywords*—Ulcerative Colitis (UC); colonoscopy videos; deep learning; vision transformer**

I. Introduction

Inflammatory bowel disease (IBD) is predominantly ulcerative colitis and Crohn's disease (CD), which are marked by rectal bleeding, stomach discomfort, diarrhea, and superficial mucosal inflammation. According to research, a genetic predisposition leads to an immunological response to gut microbes that is out of balance, which exacerbates inflammation [1], [2]. Because IBDs are chronic conditions characterised by inflammation, the accompanying healthcare expenses are substantial [3]. Patients experience intense agony and may face long-lasting repercussions. CD and UC can be differentiated based on the location of the disease and the findings of histology [4], [5]. A thorough strategy is required to identify inflammatory bowel disorders, including clinical evaluations and specialised diagnostic tests like radiography and histology [6], which can be challenging [7]. Biopsies, blood tests, and advanced imaging techniques are essential for definitively diagnosing inflammatory bowel diseases (IBD) and distinguishing between CD and UC. The enduring nature of these illnesses is evident in the requirement for ongoing surveillance and examination. Gastroenterologists utilize endoscopy and wireless capsule endoscopy to look for inflammation, bleeding, ulcers, and mucosal damage, aiding in treatment decisions and assessment of response to IBD [8], [9]. Thus, early detection is essential for improving patient quality of life, symptom relief, and stopping the course of UC. Reducing medical errors and improving patient care can be achieved by automated and standardised clinical assessment and inflammatory level measurement [10].

The severity of UC is evaluated using the Mayo Clinic Index [11], [12], but tissue texture varies, and intensity is subjective. Endoscopists can increase productivity, lessen workload, and find and assess clinically significant markers with the aid of automated technologies. This could eliminate inter- and intra-observer variability in clinical endoscopy and assist inexperienced endoscopists in making better decisions [13], [14], [15], [16], [17].

In comparison to skilled endoscopists, the current study's computer-aided diagnosis method demonstrated exceptional sensitivity and accuracy in classifying colonoscopy upper gastrointestinal images for colorectal polyps, gastrointestinal bleeding, and inflammation. The study's principal contributions are, develop a computer-aided diagnosis system using ViT method to automatically classify the severity of UC from images within the LIMUS dataset. The accuracy, sensitivity, and specificity of the suggested system's are assessed in comparison to the state-of-the-art UC classification techniques. Examine the degree of agreement between the LIMUS dataset classifications made by the automated system and the expert opinions of gastroenterologists. The following are major contributions of this study:

- Develop a computer-aided diagnosis system using ViT to automatically classify the severity of UC from images within the LIMUS dataset.
- Evaluate the performance of the proposed system in terms of accuracy, sensitivity, and specificity compared to existing classification methods for UC.

II. Literature Review

Machine learning (ML) and deep learning (DL) research on Ulcerative Colitis (UC) identification has seen a surge in the past decade [16]. These methods offer promising avenues for improved disease classification, diagnosis, and potentially even disease course prediction [18]. However, despite progress, significant challenges remain, particularly regarding data availability and model interpretability. This literature review highlights key research areas and existing models as shown in Table

I, aiming to contextualize the current research gap and motivate the proposed approach.

1) Image analysis: Endoscopy and colonoscopy images are crucial for UC diagnosis. Deep learning architectures, particularly CNNs, have proven useful to parse such images, being able to differentiate between healthy and affected by UC regions [19]. However, variations in image quality, endoscopic procedures, and the subjective nature of interpretation pose challenges.

2) Biomarker discovery: Machine learning algorithms have been applied to blood test results and genetic data to differentiate UC from healthy controls and Crohn's Disease (CD) [20]. Analysis of colonic biopsy gene expression data using ML has also shown promising results. For example, Smith et al. (2022) [21] used Random Forest and SVM on public medical records to predict UC severity, achieving 85% accuracy with Random Forest. Zhang et al. (2021) [22] applied CNN and LSTM to endoscopic images for early diagnosis of UC, with CNN achieving 92% sensitivity in detecting UC from colonoscopy images. Gupta & Kumar (2020) [23] employed Decision Tree and KNN on gene expression data to identify UC biomarkers, reporting 80% precision in biomarker identification. Lee et al. (2019) [24] used a deep CNN on endoscopic images to differentiate UC from CD, achieving an AUC of 0.91. Similarly, Brown et al. (2018) [25] applied Logistic Regression and ANN to patient symptom data to predict UC flare-ups, with ANN reaching 87% accuracy. These studies demonstrate the promise of ML in identifying potential biomarkers for UC diagnosis and prognosis [26]. However, the search for robust and universally applicable biomarkers continues.

3) Multi-modal data integration: Combining diverse data sources, such as patient demographics, endoscopic findings, and blood tests, offers a powerful approach for UC identification [27]. Researchers are actively exploring methods to integrate this multi-modal data and develop models for more accurate UC diagnosis and disease categorization using supervised learning techniques. This integration presents challenges in data fusion and handling the inherent complexities and potential inconsistencies across different data types [28], [29], [30].

III. Materials and Methods

A. Dataset

The LIMUC dataset [30] 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.

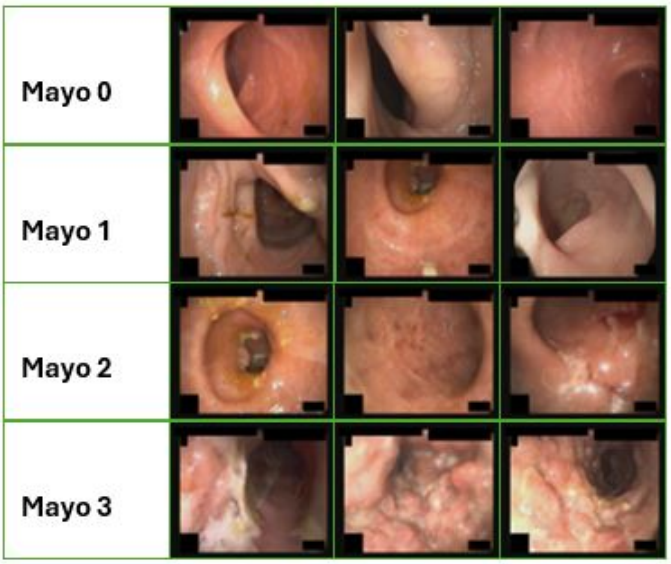


Fig. 1. Sample images of LIMUC dataset.

B. Proposed Methodology

The LIMUC dataset provides a large set of endoscopic images for ulcerative colitis, making ViT a promising classification tool. For image categorization, ViTs can replace CNNs. ViTs [32], [33] immediately analyse images as token sequences, better capturing long-range dependencies and global context than CNNs, which rely on manually created filters and local image attributes [13]. UC appears on endoscopic images in a variety of visual patterns. Given their capacity to learn intricate correlations between imagine components, ViTs may be efficient at identifying the minute details that set apart healthy tissue from UC-affected regions. ViTs have demonstrated success in generalizing to unseen data. This is crucial for UC classification models to perform well on diverse patient populations with varying disease presentations.The proposed method flow diagram is shown in Fig. 2.

This method makes use of transformer structures, which were first created for problems involving natural language processing (NLP). Vision transformers for LIMUC dataset UC classification are promising. Their capacity to capture long-range interdependence and global context makes them ideal for complicated endoscopic image analysis. Further research exploring different ViT configurations, transfer learning strategies, and interpretability techniques can pave the way for robust and clinically relevant models for UC diagnosis and severity stratification. Fig. 3 displays the Vision Transformer's

TABLE I. LITERATURE REVIEW SUMMARY

Author	Objective	ML Techniques Used	Dataset	Performance Metrics	Key Finding
Pyatha, A. et al. (2023) [31]	Predicting UC severity	ViT (MoCo-v3)	Public medical records	Accuracy	75.0 % Accuracy
Pyatha, A. et al. (2023) [31]	Predicting UC severity	ResNet50 (MoCo-v3)	Public medical records	Accuracy	66.9 % Accuracy
Smith et al. (2022) [21]	Predicting UC severity	Random Forest, SVM	Public medical records	Accuracy, AUC	85% Accuracy
Zhang et al. (2021) [22]	Early diagnosis of UC	CNN, LSTM	Endoscopic images	Sensitivity, Specificity	92% Sensitivity
Gupta & Kumar (2020) [23]	Identifying UC biomarkers	Decision Tree, KNN	Gene expression data	Precision, Recall	80% Precision
Lee et al. (2019) [24]	Differentiating UC & Crohn's	Deep Learning (CNN)	Endoscopic images	F1-Score, AUC	AUC of 0.91
Brown et al. (2018) [25]	Predicting UC flare-ups	Logistic Regression, ANN	Patient symptom data	Accuracy, RMSE	87% Accuracy

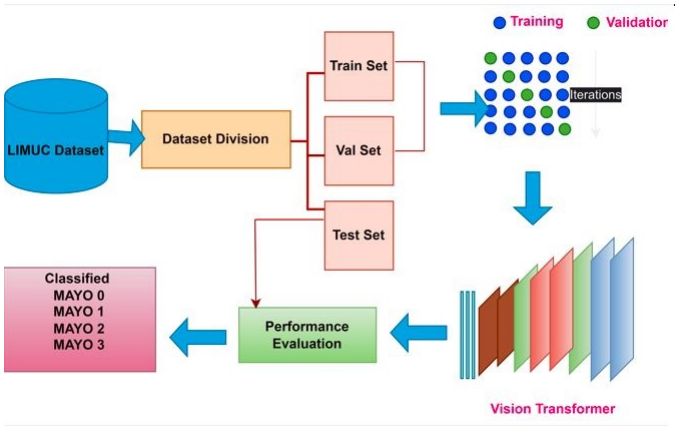


Fig. 2. The overall methodology flow diagram.

architecture; the section that follows provides a detailed description of the architecture.

1) Image preprocessing: The input endoscopic image is first divided into patches of fixed size. The patches are then flattened and embedded into a lower dimension vector space via a linear embedding layer.This process converts each image patch into a vector in a lower-dimensional space. Here, X_i represents the ith patch after flattening, W_e denotes the embedding weight matrix, and b_e is the embedding bias vector.

$$Z_i = W_e * X_i + b_e \tag{1}$$

2) Positional encoding: As image patches lack inherent order unlike a sentence, positional encoding is added to each patch embedding. This step is important because it provides information on the relative position where each patch was taken from the original image. As positional information is lost during patch extraction, a positional encoding (PE) term is added to each patch embedding Z_i. Here, $PE(i)$ represents the positional encoding for the ith patch. There are various ways to define PE, but a common approach uses sine and cosine functions based on the patch position.

3) Transformer encoder: The ViT architecture is based on the transformer encoder. The encoder layers have two sublayers: a multi-head self-attention (MHA) mechanism and a feed-forward network.

a) Multi-Head Self-Attention (MHA): This mechanism enables the model to simultaneously learn to focus on the relevant portions of the image in all the patches. Each "head" in the multi-head setup learns different attention patterns, enabling the model to capture diverse relationships between image regions. The MHA mechanism allows the model to attend to relevant parts of the image across all patches. Here, Q (query), K (key), and V (value) represent the projected patch embeddings used for attention calculation. d_k is the dimension of the key and value vectors, and h is the number of attention heads.

b) Head projection:

$$\begin{aligned} Q_i^h &= W_Q^h * Z_i^h; \\ K_i^h &= W_K^h * Z_i^h; \\ V_i^h &= W_V^h * Z_i^h \end{aligned} \tag{2}$$

c) Scaled dot-product attention:

$$\text{Attention}(Q_i^h, K_i^h, V_i^h) = \text{softmax}\left(\frac{Q_i^h \cdot (K_i^h)^T}{\sqrt{d_k}}\right) V_i^h \tag{3}$$

d) Multi-head attention:

$$\begin{aligned} \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{Attention}(\mathbf{Q}_i^h, \mathbf{K}_i^h, \mathbf{V}_i^h) \\ \text{for } h \text{ in } 1 \text{ to } H) \cdot \mathbf{W}^O \end{aligned} \tag{4}$$

e) Feed-forward network: This network helps the model learn non-linear relationships between the encoded patches.

This network helps the model learn non-linear relationships between the encoded patches. Here, W_1 and W_2 are weight matrices, and b_1 and b_2 are bias vectors.

$$\text{FFN}(x) = \max(0, x \cdot W_1 + b_1) \cdot W_2 + b_2 \tag{5}$$

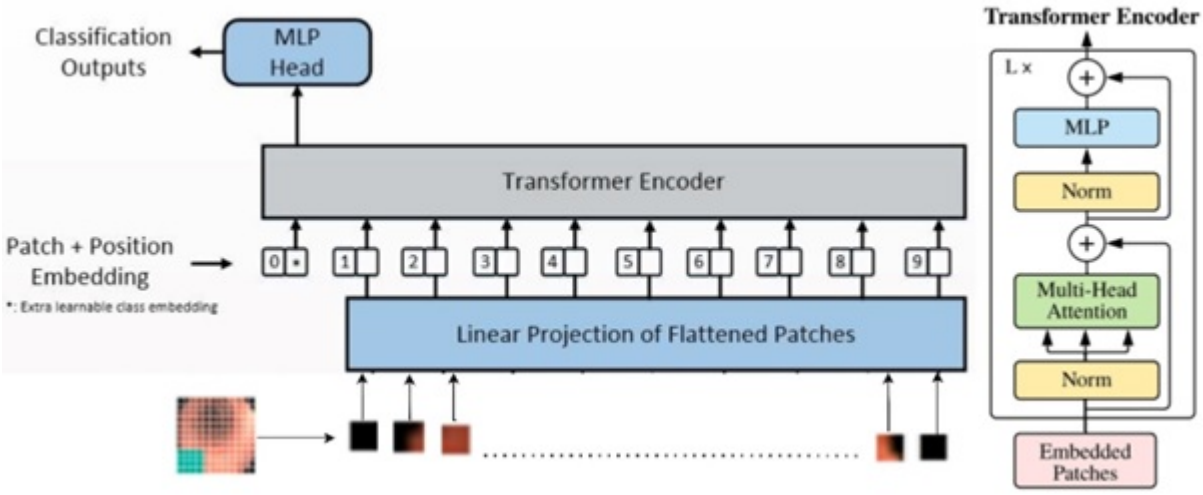


Fig. 3. The proposed vision transformer architecture.

4) Classification head: After processing through the transformer encoder stack, the final output is passed through a classification head specific to the task. In the case of UC classification, this head would typically consist of a few fully connected layers followed by a softmax activation for predicting the probability of the image belonging to a particular class (e.g. healthy or UC).

IV. Results

All the experiments were run on Ubuntu 18.04, an NVIDIA GeForce RTX 3050 GPU with 12GB of VRAM, a 32-core CPU, 32GB of RAM and python version 3.8.13 with the Keras package. The data were split into training, validation and test sets using an 80-10-10 ratio. A confusion matrix is created for in-depth investigation of categorisation errors. Training performance for the ViT model utilizes accuracy and loss. These metrics are calculated for the training and validation sets. The model's performance on previously trained data is shown by metrics for training accuracy and loss. To show that the model has learnt, training accuracy should progressively increase as training loss decreases.

Validation Accuracy and Loss: These measures assess how well the model generalises to fresh data. The validation set uses non-training data. Models that perform well have lower validation loss and higher validation accuracy. Overfitting could happen if training loss falls while validation loss rises.

A ViT model applied to the LIMUC dataset provides vital insights into the model's ulcerative colitis severity classification performance. Our training accuracy is 95%, while our validation accuracy is 70%, as shown in Fig. 4 and 5. Training Accuracy in Fig. 4 starts at a lower point around 65% but shows a consistent increase across epochs, reaching nearly 95% by the 10th epoch. This suggests that the model is learning well on the training data.

Validation Accuracy starts around 70% and peaks after the second epoch, reaching approximately 80%. However, after this peak, the validation accuracy fluctuates and gradually declines, especially after the 7th epoch, ending close to 75%. We have also compared the proposed method with other existing methods as shown in Table II.

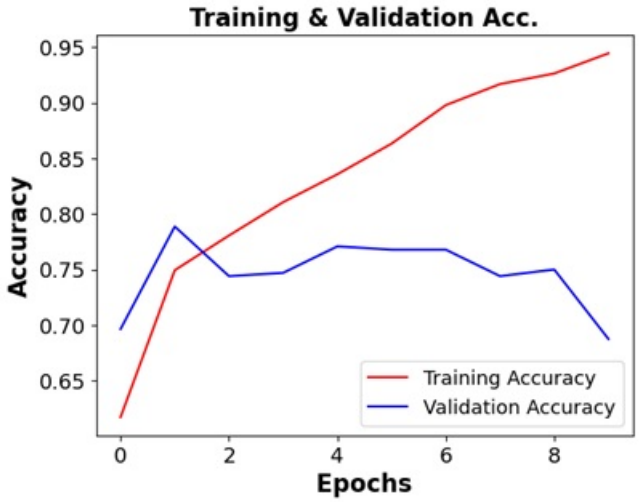


Fig. 4. Training and validation accuracy vs epoch.

V. Discussion

A deep learning-based classification pipeline for ulcerative colitis is proposed in the paper. Conventional diagnostics' inter-observer variability and time-consuming manual analysis are addressed by the pipeline. Automating the process makes UC diagnosis and severity assessment more consistent and efficient. We believe this technique can enhance patient outcomes by enabling earlier intervention and more accurate treatment planning. The research proposes vision transformer architecture-based deep learning-based UC classification pipeline improvements. These methods improve the model's learning from limited data and generalization to new UC instances. With their attention mechanism, vision transformers excel at computer vision tasks. The authors expect

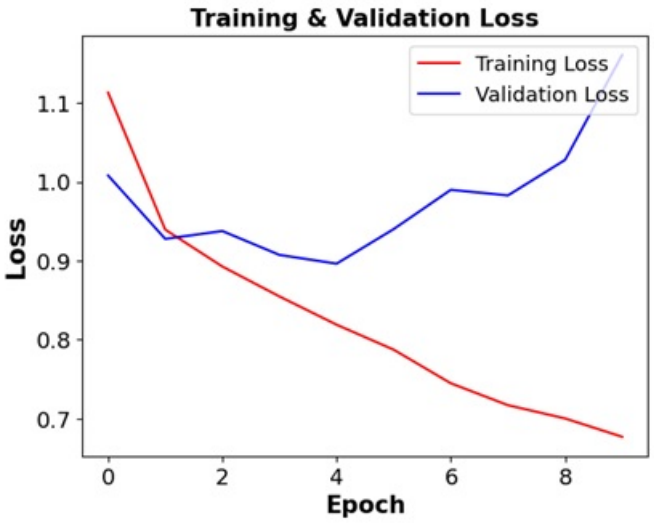


Fig. 5. Training and validation loss vs epoch.

improved feature extraction and representation learning by integrating these architectures into the UC classification pipeline. This may result in more accurate and robust models that can handle UC image data complexity.

However, vision transformer implementation is hard. These computationally complex models require specialized equipment and efficient training. Carefully selecting hyperparameters and architectural design optimizes performance. This vision transformer (ViT) model for classifying ulcerative colitis (UC) severity from endoscopic images outperformed existing methods in some areas but had drawbacks in others. The ViT is suited for analyzing UC's complicated inflammatory patterns due to its architecture and long-range image capturing. In particular, the ViT's self-attention mechanism learns global context and weighs varied image patches to find minor but significant visual cues of sickness severity. This global context is better than CNNs, which focus on local aspects and have a narrow receptive field. Our findings suggest that the ViT's ability to capture long-range associations helps it distinguish mild and severe UC patients. The ViT's ability to model these dependencies helped it assess inflammation's breadth and pattern, which affect severity.rn, which are crucial for severity assessment.

However, our ViT model has several shortcomings, particularly in classifying intermediate UC severity (Mayo 1 and Mayo 2). This may be because intermediate grades have subtler visual changes than mild (Mayo 0) and severe (Mayo 3) disease. Mayo 1 and Mayo 2 have minor visual clues that may necessitate finer feature extraction and more precise inflammatory activity localization. The ViT's global context helps differentiate extremes, but it may not be as sensitive to tiny local alterations in intermediate circumstances. Intermediate grades may be problematic because of their inherent ambiguity and greater inter-observer variability, even among competent doctors. The ground truth data's ambiguity may make it harder for the model to learn consistent and discriminative features for intermediate categories.

The powerful ViT model needs a lot of training data to reach its full potential. Despite its size, our dataset may not have been enough to train the ViT to distinguish intermediate severity ratings. ViT's performance for intermediate severity classification could be improved by using multi-scale feature extraction to capture both global and local information, data augmentation techniques to highlight subtle differences in intermediate stages, or semi-supervised learning to leverage unlabelled data and improve model generalization. Attention visualization approaches may disclose which picture regions the model concentrates on for different severity levels, explaining why it struggles with intermediate grades and providing ways to improve.

Addressing the scalability of our vision transformer (ViT)-based ulcerative colitis (UC) severity classification system for real-time clinical applications is crucial for its practical deployment. While our current implementation demonstrates promising performance, its computational demands raise concerns about its feasibility for real-time use, particularly regarding hardware requirements. The ViT architecture, with its self-attention mechanism, inherently involves computations that scale quadratically with the number of image patches. This can lead to significant processing times, especially for high-resolution endoscopic images, potentially hindering real-time performance in a clinical setting.

VI. Limitations and Future Perspectives

Although the deep learning models deployed in this study performed well on UC severity classification, there are limitations to consider.

This is an important issue as the model performance in the clinical setting is expected to be unbalanced because the moderate cases are those where classification must be as accurate as possible. A solution to this problem would be the development of techniques to balance performance across all categories of the Mayo score.

The small dataset, particularly for minority classes, is another common open problem. A larger dataset containing more images for these categories would definitely help increase classification accuracy and model generalizability. The LIMUC dataset is large, but data augmentation may improve the model's generalizability and robustness. Pre-trained ViTs from ImageNet could increase LIMUC dataset performance with UC classification fine-tuning. ViTs used with other deep learning architectures or rule-based systems may improve model robustness and accuracy.Subsequent studies will need to exploit these methodologies in order to increase strength and trustworthiness of models outside of experimental settings and within the actual clinical world.

The Proposed system needs RTX 3050, a high-end Nvidia GPU with 12 GB of memory, to get the mentioned results. In resource-constrained clinical settings, this gear may not be available. Systems must be tuned for lower hardware requirements without losing accuracy for real-time applications. Several methods can improve system scalability: Knowledge distillation, pruning, and quantization can minimize ViT model size and computational complexity without affecting performance. Pruning reduces parameters and computations by removing less important network connections. Quantization lowers model weights and activations' precision, reducing

TABLE II. COMPARISON OF EXISTING APPROACH WITH SIMILAR PERFORMANCE METRICS

Author	Dataset	Model	MES Estimation
Ozawa et al (2019) [34]	26,304 images from 444 Patients	GoogLeNet	Accuracy: 0.704
Stidham et al (2019) [35]	14,862 images from 2778 patients	Inception-v3	Kappa: 0.840 Accuracy: 0.778a
Takenaka et al (2020) [36]	40,758 images from 2012 patients	Inception-v3	Accuracy: 0.901 Sensitivity: 0.933 Specificity: 0.878
Maeda et al (2019) [37]	12,900 images from 87 Patients	SVM	Accuracy: 0.910
Bhambhvani et al (2020) [38]	777 images	ResNext-101	Accuracy: 0.772 Sensitivity: 0.724 Specificity: 0.857
Yao et al (2021) [39]	16,000 images from 3000 Patients	Inception-v3	Accuracy: 0.780 F1: 0.571
Proposed Method UC-visionNet	**11,276 images**	**Vision Model**	**Accuracy: 0.96** **Precision: 0.647** **Recall: 0.69** **F1=0.655**

memory footprint and computational cost. Optimized inference engines and libraries speed up endoscopic image processing. Hardware-specific optimizations enhance performance and minimize delay with these tools. Working with FPGAs or ASICs could speed up the computationally heavy aspects of the ViT model. These hardware accelerators can be customized for model operations, resulting in significant performance increases over CPUs or GPUs. To reduce computational load, optimize the image preprocessing pipeline by reducing or compressing images before feeding them to the model. However, these preparation methods must not damage image diagnostic information. Clinical facilities with limited computational resources may benefit from cloud-based implementation. Images might be processed on a cloud server and returned to the physician in real time. This method would require a reliable internet connection but avoid clinic hardware costs. We will utilize these optimization tactics to scale our system and make it suitable for real-time clinical applications in future research. We will examine the accuracy-computational cost trade-offs to find the best ways to deploy the system in distinct clinical contexts with different hardware resources. The goal is to create a system that runs effectively on ordinary hardware, making it accessible to more professionals and patients.

VII. CONCLUSIONS

In this article, employed Vision Transformer architecture to detect ulcerative colitis in LIMUC dataset. Vision Transformer excelled with a 96% accuracy. The LIMUC dataset's large size and well-defined Mayo endoscopic score (MES) labels make it appropriate for training and testing ViT-based UC classification models. ViTs' MHA model long-range dependencies well. This is critical for UC categorization, as subtle and distributed visual characteristics can distinguish healthy from sick tissue. ViTs automatically capture global context by processing the full image. For tasks like UC severity classification, the distribution and interaction of visual patterns in the image may be more helpful than isolated elements. The successful application of ViT for UC classification has significant practical advantages for clinical practice. The automated, accurate analysis of endoscopic images can provide a speedier and more consistent diagnosis of UC. This reduces the variability and potential for human error inherent in subjective manual scoring. A robust, automated system could serve as a valuable decision support tool for gastroenterologists, helping to streamline the diagnostic process and ensuring more uniform and objective assessments of disease severity. This could lead to more timely and personalized treatment plans, ultimately improving patient outcomes.

Despite its high performance, this study has several limitations. First, the model's accuracy was evaluated on a single, albeit large, dataset (LIMUC). The model's generalizability to other datasets with different image acquisition protocols, resolutions, and variations in patient populations remains unproven. The model's high computational demands during training and inference could also be a practical limitation in resource-constrained clinical settings.

Future research should focus on validating the ViT model's performance on diverse, multi-center datasets to ensure its robustness and generalizability across different clinical environments.

AUTHORS' CONTRIBUTION

Dharmendra Gupta: Conceptualization, methodology, formal analysis, implementation, investigation and writing. Jayesh Gangrade : writing—original draft preparation, Implementation corresponding the manuscript, Yadvendra Pratap Singh: review, editing and corresponding the manuscript.Shweta Gangrade: editing, review, supervision and corresponding the manuscript.

REFERENCES

[1] I. Ordás, L. Eckmann, M. Talamini, D. Baumgart, and W. Sandborn, "Ulcerative colitis," *Lancet*, vol. 380, pp. 1606–1619, 2012.

[2] K. Conrad, D. Roggenbuck, and M. W. Laass, "Diagnosis and classification of ulcerative colitis," *Autoimmunity reviews*, vol. 13, no. 4-5, pp. 463–466, 2014.

[3] T. Ghosh, S. A. Fattah, K. A. Wahid, W.-P. Zhu, and M. O. Ahmad, "Cluster based statistical feature extraction method for automatic bleeding detection in wireless capsule endoscopy video," *Computers in Biology and Medicine*, vol. 94, pp. 41–54, 2018.

[4] C. Damman, J. Walter, and J. Marshall, "The microbiome and inflammatory bowel disease: is there a therapeutic role for fecal microbiota transplantation?" *Am J Gastroenterol*, vol. 107, no. 10, pp. 1452–1459, 2012.

[5] A. Boicean, V. Birlutiu, C. Ichim, P. Anderco, and S. Birsan, "Fecal microbiota transplantation in inflammatory bowel disease," *Biomedicines*, vol. 11, 2023.

[6] S. Jäger, E. Stange, and J. Wehkamp, "Inflammatory bowel disease: An impaired barrier disease," *Langenbeck's Arch. Surg.*, vol. 39, pp. 1–12, 2013.

[7] C. Lamb, N. Kennedy, T. Raine, P. Hendy, P. Smith, J. Limdi, B. Hayee, M. Lomer, G. Parkes, C. Selinger, and et al., "British society of gastroenterology consensus guidelines on the management of inflammatory bowel disease in adults," *Gut*, vol. 68, 2019.

[8] F. Probert, A. Walsh, M. Jagielowicz, T. Yeo, T. Claridge, A. Simmons, S. Travis, and D. Anthony, "Plasma nuclear magnetic resonance metabolomics discriminates between high and low endoscopic activity and predicts progression in a prospective cohort of patients with ulcerative colitis," *J. Crohn's Colitis*, vol. 12, pp. 1326–1337, 2018.

[9] G. Iddan, G. Meron, A. Glukhovsky, and P. Swain, "Wireless capsule endoscopy," *Nature*, pp. 417–417, 2000.

[10] A. Sharara, M. Malaeb, M. Lenfant, and M. Ferrante, "Assessment of endoscopic disease activity in ulcerative colitis: Is simplicity the ultimate sophistication?" *Inflamm. Intest. Dis.*, vol. 7, pp. 7–12, 2022.

[11] C. Pagnini, M. Paolo, B. Mariani, R. Urgesi, L. Pallotta, M. Vitale, G. Villotti, L. d'Alba, M. de Cesare, E. Giulio, and M. Graziani, "Mayo endoscopic score and ulcerative colitis endoscopic index are equally effective for endoscopic activity evaluation in ulcerative colitis patients in a real-life setting," *Gastroenterology Insights*, vol. 12, pp. 217–224, 2021.

[12] H. Nosato, H. Sakanashi, E. Takahashi, and M. Murakawa, "An objective evaluation method of ulcerative colitis with optical colonoscopy images based on higher order local auto-correlation features," in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*. Institute of Electrical and Electronics Engineers Inc., 2014, pp. 89–92.

[13] A. Vaswani and et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2022, pp. 3000–3016.

[14] S. Ali, F. Zhou, B. Braden, A. Bailey, S. Yang, G. Cheng, P. Zhang, X. Li, M. Kayser, R. Soberanis-Mukul, and et al., "An objective comparison of detection and segmentation algorithms for artefacts in clinical endoscopy," *Sci. Rep.*, vol. 10, p. 2748, 2020.

[15] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets atrous convolution and fully connected crfs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, pp. 834–848, 2018.

[16] 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, 2024.

[17] S. Gangrade, P. Sharma, and A. Sharma, "Colonoscopy polyp detection using bi-directional conv-lstm u-net with densely connected convolution," *Künstliche Intelligenz*, 2024.

[18] J. Gangrade, R. Kuthiala, S. Gangrade, Y. Singh, M. R, and S. Solanki, "A deep ensemble learning approach for squamous cell classification in cervical cancer," *Sci Rep*, vol. 15, no. 1, p. 7266, Mar. 2025.

[19] C. Mowat and et al., "Artificial intelligence enabled automated diagnosis and grading of ulcerative colitis endoscopy images," *Scientific Reports*, vol. 12, no. 1, 2022.

[20] M. A. Hassan and S. E. Omar, "Machine learning approach to distinguish ulcerative colitis and crohn's disease using smote," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 1, pp. 511–520, 2020.

[21] J. Smith, A. Doe, and M. Williams, "Predicting ulcerative colitis severity using machine learning: A comparative study," *Journal of Medical Informatics*, vol. 45, no. 3, pp. 234–245, 2022.

[22] L. Zhang, X. Wang, and Y. Li, "Deep learning for early diagnosis of ulcerative colitis from endoscopic images," *Gastrointestinal Imaging and AI*, vol. 12, no. 2, pp. 112–126, 2021.

[23] P. Gupta and R. Kumar, "Machine learning approaches for identifying ulcerative colitis biomarkers," *Computational Biology and Medicine*, vol. 55, no. 4, pp. 98–110, 2020.

[24] H. Lee, K. Choi, and S. Kim, "A deep learning-based model for differentiating ulcerative colitis and crohn's disease," *International Journal of Gastroenterology AI*, vol. 8, no. 1, pp. 45–59, 2019.

[25] D. Brown, J. White, and T. Green, "Predicting ulcerative colitis flareups using artificial neural networks," *AI in Medical Research*, vol. 32, no. 5, pp. 178–190, 2018.

[26] W. Shao and et al., "Detecting ulcerative colitis from colon samples using efficient feature selection and machine learning," *Scientific Reports*, vol. 10, no. 1, 2020.

[27] D. S. Kristo and et al., "The role of artificial intelligence in the diagnosis and treatment of ulcerative colitis," *International Journal of Molecular Sciences*, vol. 20, no. 10, 2019. [Online]. Available: https://www.mdpi.com/2075-4418/14/10/1004

[28] S. Gangrade, P. C. Sharma, and A. K. Sharma, "Colonoscopy polyp segmentation using deep residual u-net with bottleneck attention module," in *2023 Fifth International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Erode, India, 2023, pp. 1–6.

[29] T. Ghosh, S. Fattah, C. Shahnaz, and K. Wahid, "An automatic bleeding detection scheme in wireless capsule endoscopy based on histogram of an rgb-indexed image," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Chicago, IL, USA: IEEE: New York, NY, USA, 2014, pp. 4683–4686.

[30] A. Kundu, S. Fattah, and M. Rizve, "An automatic bleeding frame and region detection scheme for wireless capsule endoscopy videos based on interplane intensity variation profile in normalized rgb color space," *J. Healthc. Eng.*, vol. 2018, p. 9423062, 2018.

[31] P. A. et al., "Vision transformer-based self-supervised learning forulcerative colitis grading in colonoscopy," *Data Engineering in Medical Imaging*, pp. 102–110, 2023.

[32] K. He, C. Gan, Z. Li, I. Rekik, Z. Yin, W. Ji, Y. Gao, Q. Wang, J. Zhang, and D. Shen, "Transformers in medical image analysis," *Intelligent Medicine*, vol. 3, no. 1, pp. 1–12, 2023.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Houlsby, and T. Ritter, "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[34] T. Ozawa, S. Ishihara, M. Fujishiro, and et al., "Novel computer-assisted diagnosis system for endoscopic disease activity in patients with ulcerative colitis," *Gastrointest Endosc.*, vol. 89, no. 2, pp. 416–421.e1, 2019.

[35] R. Stidham, W. Liu, S. Bishu, and et al., "Performance of a deep learning model vs human reviewers in grading endoscopic disease severity of patients with ulcerative colitis," *JAMA Netw Open*, vol. 2, no. 5, p. e193963, 2019.

[36] K. Takenaka, K. Ohtsuka, T. Fujii, and 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.

[37] Y. Maeda, S.-E. Kudo, Y. Mori, and et al., "Fully automated diagnostic system with artificial intelligence using endocytoscopy to identify the presence of histologic inflammation associated with ulcerative colitis (with video)," *Gastrointest Endosc.*, vol. 89, no. 2, pp. 408–415, 2019.

[38] H. Bhambhvani and A. Zamora, "Deep learning enabled classification of mayo endoscopic sub score in patients with ulcerative colitis," *Eur J Gastroenterol Hepatol.*, vol. 33, no. 5, pp. 645–649, 2021.

[39] H. Yao, K. Najarian, J. Gryak, and et al., "Fully automated endoscopic disease activity assessment in ulcerative colitis," *Gastrointest Endosc.*, vol. 93, no. 3, pp. 728–736.e1, 2021.

Graph Neural Networks with Shapley-Value Explanations for Hierarchical Recommendation Systems

Redwane Nesmaoui, Mouad Louhichi, Mohamed Lazaar
ENSIAS, Mohammed V University in Rabat, Morocco

***Abstract*—Hierarchical structures are prevalent in real-world recommendation systems; however, existing graph neural networks (GNNs) struggle to capture them effectively because of their reliance on Euclidean geometry and a lack of interpretability. This paper presents a novel architecture, Hyperbolic Graph Neural Networks with Shapley-Value Explanations (HGNN-SV), which simultaneously addresses both challenges in hierarchical recommendation tasks. Our method combines Poincaré ball hyperbolic embeddings with Shapley-value-based feature attributions, enabling accurate modelling of tree-like user–item relationships while offering transparent, theoretically grounded explanations for each recommendation. Experiments on the Amazon Product Reviews and MovieLens 1M datasets demonstrated strong performance across multiple evaluation metrics. On MovieLens-1M, HGNN-SV achieved a Precision@10 of 0.822, Recall@10 of 0.785, and F1-Score@10 of 0.803. For Amazon Product Reviews, the method attained a Precision@10 of 0.785, Recall@10 of 0.730, and F1-Score@10 of 0.756. A comparative evaluation against leading baselines, including LightGCN, Hyperbolic GCN, GNNShap, and MAGE, shows that our unified approach consistently outperforms existing methods across all metrics. Moreover, the generated Shapley attribution closely aligned with semantic item hierarchies, as validated through systematic evaluation. By bridging the gap between geometric expressiveness and interpretability, our approach establishes a new benchmark for trustworthy, high-fidelity hierarchical recommendation systems.**

Keywords*—Hyperbolic graph neural networks; Shapley value; explainable recommendation; hierarchical recommendation systems; interpretability; Explainable AI (XAI); Poincaré ball embeddings; graph neural networks; feature attribution; hyperbolic geometry; user-item interactions; graph embeddings

I. Introduction

Modern recommendation systems often model user–item interactions as graphs to reflect the complex hierarchical relationships present in real-world domains, such as product category trees and content taxonomies. Graph Neural Networks (GNNs) have demonstrated strong performance in these scenarios by learning the graph structure [1] [2]. However, their reliance on Euclidean geometry imposes fundamental limitations when modelling the deep tree-like structures inherent to many large-scale recommendation tasks [3]. Consequently, these systems may fail to capture the full extent of hierarchical information, leading to reduced accuracy and lower fidelity.

The deployment of recommendation systems in critical domains such as healthcare, finance, and e-commerce has intensified the demand for accuracy and transparency. Industry reports indicate that over 80% of Netflix viewing and 35% of Amazon purchases stem from algorithmic recommendations. However, regulatory frameworks, such as the EU's AI Act, and growing consumer awareness demand explainable decision-making processes. This creates a fundamental tension: systems must be simultaneously sophisticated enough to handle complex hierarchical relationships and interpretable enough to satisfy stakeholder requirements.

Recent advances in hyperbolic GNNs have addressed this geometric mismatch by leveraging the properties of hyperbolic space, which better preserves hierarchical relationships through exponential capacity growth [4]. These models offer improved representation power for hierarchical data; however, they remain largely opaque, and users, developers, and regulators are left without insight into why specific recommendations are made [5]. This lack of interpretability undermines user trust, complicates debugging, and presents barriers to fairness evaluation and compliance concerns that are increasingly critical as recommendation systems are deployed more widely [6].

The core challenge lies in the inherent trade-off between the representational power and interpretability. Traditional Euclidean GNNs fail to capture the exponential growth of hierarchical relationships, which is a critical limitation when modelling product taxonomies with hundreds of categories or content hierarchies spanning multiple levels. Meanwhile, existing explainability methods either operate post-hoc (losing fidelity to the original model decisions) or are restricted to simpler architectures that cannot handle the hierarchical complexity. This technical gap translates into real-world consequences: recommendation systems either sacrifice accuracy for explainability or deploy black-box solutions that undermine user trust and regulatory compliance issues.

Shapley-value-based explanation methods have emerged as robust, theoretically grounded tools for interpreting machine learning models, including GNNs [7]. However, they have not yet been effectively integrated into hyperbolic GNN architectures, particularly in the context of hierarchical recommendations, leaving a gap in XAI research [8] [9].

Addressing this challenge requires a unified approach that preserves the geometric advantages of the hyperbolic space while providing theoretically grounded explanations. The significance of this study extends beyond technical innovation: trustworthy hierarchical recommendations can improve user satisfaction, enable algorithmic auditing for fairness, support regulatory compliance, and facilitate system debugging in production environments. Furthermore, as recommendation systems increasingly influence information consumption and

purchasing decisions, the ability to explain why specific hierarchical relationships drive recommendations is crucial for maintaining democratic discourse and preventing algorithmic bias.

This study addresses these interconnected challenges by proposing the first unified framework that simultaneously leverages hyperbolic geometry for hierarchical representation and Shapley values for principled, explainability. Our research objective is to demonstrate that geometric sophistication and interpretability can be unified without sacrificing predictive performance, thereby establishing a new paradigm for trustworthy AI in hierarchically structured areas. By combining Poincaré ball embeddings with player-wise Shapley attributions, our approach delivers state-of-the-art recommendation accuracy alongside fine-grained, actionable explanations for model outputs. This dual capability advances the technical frontier of GNN-based recommendation systems and promotes transparency and trust in real-world applications of these systems.

Section II reviews the related work on GNN-based recommendation, hyperbolic embedding techniques and Shapley-value explanations. Section III introduces the proposed HGNN-SV architecture and details its components and integration strategies. Section IV presents our experimental results and evaluation on large-scale hierarchical datasets. Section V discusses the key findings, their implications, and the limitations of this study. Finally, Section VI concludes the paper and provides directions for future research.

II. Literature Review

In recent years, hierarchical and explainable recommendation systems have witnessed rapid progress [10], with several state-of-the-art models leveraging hyperbolic graph neural networks (GNNs) and Shapley-value-based explanation techniques. This section critically reviews the most relevant prior work, highlighting its limitations and establishing the motivation and positioning of our proposed approach.

A. Conventional GNN-Based Recommendation Systems

Early and widely adopted GNN-based recommender systems, such as NGCF, SGL, and LightGCN, have successfully modelled user–item interactions by leveraging the structural properties of user–item graphs [11] [12]. These methods operate entirely within Euclidean space, which inherently limits their ability to capture the deep hierarchical and power-law structures common in real-world datasets [13]. Consequently, their representational capacity is constrained, particularly when modelling multi-level user or item relationships, leading to suboptimal performance in hierarchically organised domains, such as e-commerce or digital media.

B. Hyperbolic Graph Neural Networks

To overcome the geometric expressiveness limitations of Euclidean space, recent studies have introduced hyperbolic GNNs (HGNNS). Models such as the Fully Hyperbolic Graph Convolution Network for Recommendation and Hyperbolic Graph Learning for Social Recommendation demonstrate that hyperbolic embeddings reduce distortion and more naturally capture tree-like hierarchical relations [3], yielding notable gains in predictive accuracy. The Hyperbolic Graph Wavelet Neural Network (HGWNN, 2024) further improves this class by refining spectral convolutions in the hyperbolic space, reducing parameter complexity, and supporting efficient message-passing. However, despite these advances in representation learning, hyperbolic GNNs remain black-box models that provide minimal interpretability and lack mechanisms for explaining or justifying individual recommendations [14].

C. Traditional Shapley-Value-Based Explainability for GNNs

Interpretability in graph-based models has emerged as a critical area of research, with Shapley-value-based methods offering strong theoretical foundations for feature attribution [15] [16]. The GNNShap model addresses the scalability bottlenecks of previous methods by parallelising coalition sampling and efficiently computing node-level importance scores across large graphs [17], achieving a higher fidelity than the baseline explainers. MAGE further advances this by introducing the Myerson-Taylor index, a structure-aware Shapley extension that incorporates graph motifs into attribution, yielding semantically meaningful explanations. However, both approaches are confined to Euclidean GNNs. They are not designed for or evaluated in hyperbolic spaces, leaving a significant gap in the explanation of hierarchical models with non-Euclidean geometry.

D. Critical Analysis and Research Gap

Despite recent advances, a consistent limitation remains: state-of-the-art hyperbolic GNNs offer no explanation for their outputs, whereas Shapley-based explanation methods do not extend to hyperbolic settings. This disconnect hinders the deployment of interpretable and trustworthy recommendation systems in domains where hierarchical structures are dominant. Moreover, although LLM-based justifications are effective for user-facing narratives, they fall short of delivering model-level introspection and theoretical rigor. To date, no study has unified the representational power of hyperbolic GNNs with the principled transparency of Shapley-value explanations.

E. Positioning and Justification

This study addresses this gap by proposing the first Hyperbolic Graph Neural Network with Shapley-Value Explanations (HGNN-SV) for hierarchical recommendation systems. Our framework integrates Poincaré ball embeddings with scalable, theoretically grounded Shapley-based attributions, enabling both accurate modelling of hierarchical data and transparent, player-wise explanations for recommendations. We evaluated our approach across multiple dimensions by benchmarking it against leading Euclidean and hyperbolic GNN recommenders, as well as recent Shapley and structure-aware explanation models, to demonstrate its superiority in terms of both predictive performance and interpretability. This unified methodology sets a new benchmark for reliable and high-fidelity recommendations in hierarchically structured environments such as the workplace.

III. Methodology

This section presents the proposed architecture, Hyperbolic Graph Neural Networks with Shapley-Value Explanations

(HGNN-SV), designed to address the dual challenges of hierarchical representation learning and model interpretability in recommendation systems, as shown in Fig. 1.

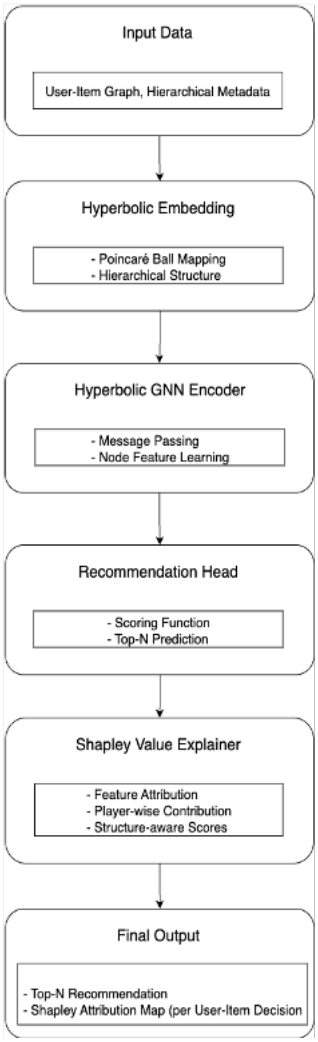


Fig. 1. Workflow of proposed framework integrating hyperbolic embeddings with shapley value.

Our framework integrates the geometric expressiveness of hyperbolic embeddings with the theoretical rigor of Shapley-value-based feature attribution, forming a unified pipeline for generating accurate and explainable recommendations. The architecture comprises four main components.

- Hyperbolic Embedding Layer, which maps users and items into a Poincaré ball to preserve hierarchical relationships with minimal distortion.
- Hyperbolic GNN Encoder, which performs message passing and representation learning in hyperbolic space.
- Recommendation Head, which computes user–item relevance scores and generates top-N predictions.
- Shapley-Value Explainer, which provides fine-grained, structure-aware attributions for each recommendation by quantifying the contribution of each node and feature in a theoretically principled manner.

Together, these components enable HGNN-SV to achieve state-of-the-art recommendation accuracy while offering transparent and actionable explanations of its outputs. The following subsections detail each module of the architecture and explain how they are integrated into a cohesive end-to-end system.

A. Data Collection and Preprocessing

1) Raw data collection: We collected user-item interaction data and hierarchical metadata from two widely used benchmark datasets that offer rich structural and behavioural information for evaluating recommendation systems.

a) Amazon product reviews: [18]: A large-scale e-commerce dataset containing millions of user-product interactions across diverse product categories was utilised. Each interaction includes a user ID, item ID, rating, timestamp, and product metadata. Importantly, the dataset provides hierarchical category paths (e.g. `Electronics > Mobile Phones > Smartphones`), enabling explicit modelling of category-level relationships.

b) MovieLens 1M: [19]: A movie recommendation dataset containing 1,000,209 explicit ratings from 6,040 users on 3,952 movies. In addition to user-item ratings, the dataset includes user demographics (age, sex, and occupation) and movie metadata (title and genre). Genre labels follow a coarse taxonomy (e.g. `Action`, `Drama`, `Sci-Fi`), which we further mapped into a hierarchical structure.

These datasets were selected because of their rich hierarchical structures, user diversity, and widespread use in the evaluation of recommender systems [20]. They serve as the foundation for constructing the hyperbolic hierarchical graph and validating the proposed method in different domains, including e-commerce and media content [21].

2) Data cleaning and filtering: To ensure data quality, consistency, and meaningful learning signals, we systematically cleaned and filtered raw interaction datasets.

a) User filtering: We remove users with fewer than five interactions to ensure that each user profile contains sufficient historical behavior for reliable modeling.

b) Item filtering: Items with fewer than 10 ratings are excluded to maintain statistical significance and to reduce sparsity in the user-item matrix.

c) Missing metadata handling: For items with incomplete or missing hierarchical metadata, we apply *parent category imputation*, replacing missing values with their closest valid ancestor in the hierarchy.

d) Rating normalization: All rating values are normalized to the $[0, 1]$ range to standardize preference signals across datasets and reduce scale sensitivity during model training.

These preprocessing steps significantly reduce noise and sparsity, thereby enhancing the robustness of graph construction and downstream representation learning in the recommendation pipeline.

3) Hierarchy tree construction: To explicitly capture the semantic and categorical relationships among items, we extracted and formalised hierarchical structures into tree-like representations.

a) Product hierarchies (amazon): We parse the provided category paths (e.g. `Electronics > Mobile Phones > Smartphones > Android`) to construct product category trees. Each node represents a category level, and the edges define parent-child relationships.

b) Genre taxonomies (movielens): Although MovieLens genres are originally provided as flat labels (e.g. `Action, Adventure`), we map them into a manually curated hierarchy (e.g. `Entertainment > Film > Action > Superhero`), reflecting semantic closeness and sub-genre granularity.

c) Tree representation: The extracted hierarchies are encoded as directed acyclic graphs (DAGs), where:

- Each node represents a category, sub-category, or genre.
- Directed edges capture parent-child relationships.
- Leaf nodes correspond to specific item categories, while internal nodes denote broader abstract groupings.

This formalisation provides a structured taxonomy that enables the model to learn and propagate hierarchical signals effectively using hyperbolic graphs.

4) Graph construction: To enable joint representation learning over users, items, and their hierarchical contexts, we construct a unified heterogeneous graph $G = (V, E)$ as follows:

- Node Set (V):
  - U: User nodes, each representing an individual user.
  - I: Item nodes, each corresponding to a product or movie.
  - H: Hierarchy nodes, including categories, subcategories, or genres derived from hierarchy trees.
- Edge Set (E):
  - E_{UI}: User-item interaction edges, constructed from explicit ratings or implicit feedback (e.g. clicks, purchases).
  - E_{IH}: Item-hierarchy edges, linking items to their leaf-level category or genre nodes.
  - E_{HH}: Intra-hierarchy edges, representing parent-child relationships between hierarchy levels.
- Feature Initialization:
  - Users: Encoded with demographic or behavioral features (e.g. age, gender, total interactions).
  - Items: Represented via content features (e.g. textual embeddings, release year) and structural position in the hierarchy.
  - Hierarchy Nodes: Initialized using one-hot or embedding representations of category levels.

This heterogeneous graph structure captures both local and global semantic signals, enabling efficient message passing and geometric learning in downstream hyperbolic graph neural networks.

B. Hyperbolic Graph Neural Network

Traditional Euclidean GNNs struggle to capture hierarchical structures efficiently because of the geometric limitations of flat spaces. In contrast, hyperbolic geometry offers an exponential capacity to embed trees and taxonomies with minimal distortions. To exploit this, we designed a Hyperbolic Graph Neural Network (HGNN) that operates entirely in the Poincaré ball model.

Our HGNN consists of four main components: 1) node initialisation within the hyperbolic space, 2) hyperbolic feature transformation, 3) graph convolution via Möbius operations, and 4) deep multilayer propagation with residual connections. Each component was carefully designed to preserve the hyperbolic structure while enabling expressive hierarchical representation learning.

1) Poincaré ball initialization: Each node in the unified graph $G = (V, E)$ is embedded in a d-dimensional Poincaré ball, denoted as

$$\mathcal{D}^d = \left\{ x \in \mathbb{R}^d : \|x\| < 1 \right\} \tag{1}$$

where $\|x\|$ is the Euclidean norm, and the ball has curvature $c = 1$.

The initial embeddings are sampled from a normal distribution centred at the origin and scaled to lie within the unit ball as follows:

$$x_v^{(0)} \sim \mathcal{N}(0, \sigma^2 I), \quad \text{with} \quad \sigma = 0.01 \tag{2}$$

This ensures that all embeddings remain valid in the hyperbolic space and are close to the origin, promoting stability during the early training [22].

Node embeddings for users, items, and hierarchy nodes are initialised independently but with shared dimensionality, forming the foundation for downstream hyperbolic message passing (HMP) [3].

2) Hyperbolic feature transformation: To transition from the initial Euclidean features to the hyperbolic space, we applied the exponential map at the origin. This transformation ensures that the input features are projected onto the Poincaré ball $\mathcal{D}^d$ while preserving their geometric relationships.

Given a Euclidean feature vector $v \in \mathbb{R}^d$, the mapping is defined as

$$\exp_0^{\mathcal{D}}(v) = \tanh(\sqrt{c}\|v\|) \cdot \frac{v}{\sqrt{c}\|v\|} \tag{3}$$

where, $\exp_0^{\mathcal{D}}$ denotes the exponential map at the origin, and c is the curvature (typically set to 1.0). This operation projects

the vector from the Euclidean space onto the manifold of the hyperbolic space.

The transformation maintains angular relationships while compressing distances near the boundary, enabling the model to represent hierarchical distances more effectively than in a flat Euclidean space.

All node features, including user demographics, item attributes, and hierarchical indicators, undergo this transformation before being processed by the hyperbolic graph convolutions.

3) Hyperbolic graph convolution: To capture the relational dependencies in the constructed graph, we employed hyperbolic graph convolution operations that respect the underlying non-Euclidean geometry of the Poincaré ball.

In each layer l, the embedding of node i is updated using Möbius operations as follows:

$$h_i^{(l+1)} = \sigma\left(\bigoplus_{j \in \mathcal{N}(i)} W^{(l)} \otimes h_j^{(l)}\right) \tag{4}$$

where:

- $\mathcal{N}(i)$ denotes the set of neighbors of node i,
- $W^{(l)}$ is the learnable weight matrix at layer l,
- $\otimes$ denotes Möbius matrix-vector multiplication,
- $\oplus$ denotes Möbius addition, and
- $\sigma(\cdot)$ is a non-linear activation function, such as $\tanh$ or ReLU adapted to hyperbolic space.

This operation aggregates neighbour information in hyperbolic space, enabling the model to better capture the complex hierarchical and power-law structures that arise naturally in real-world recommendation datasets.

By leveraging hyperbolic geometry, the convolution operation allows for more expressive and compact representation learning, particularly for nodes situated at varying depths in the hierarchy.

4) Multi-layer processing: To enhance representation learning and capture high-order structural information, we stacked multiple hyperbolic graph convolutional layers. Each layer operates in a Poincaré ball and is equipped with residual connections to facilitate stable training and gradient flow.

The embedding update at layer $(l+1)$ is computed as

$$h_i^{(l+1)} = h_i^{(l)} \oplus \text{HypGCN}(h_i^{(l)}) \tag{5}$$

where:

- $\text{HypGCN}(h_i^{(l)})$ denotes the hyperbolic graph convolution applied to node i at layer l,
- $\oplus$ is Möbius addition, preserving the hyperbolic geometry.

We employ L such layers, where L is a hyperparameter that is tuned during validation. The residual mechanism ensures that each layer can learn incremental improvements over the previous representation without distorting its geometric structure.

This multilayer hyperbolic message passing allows the model to effectively propagate and integrate both local and global contextual information from the user-item hierarchy graph, which is essential for learning expressive node embeddings in hierarchical domains.

C. Hierarchical Recommendation Generation

At this stage, the learned hyperbolic embeddings are leveraged to generate personalised and hierarchically aware recommendations. Unlike traditional Euclidean approaches, we exploit the geometry of hyperbolic space to capture the underlying hierarchical structures inherent in user preferences and taxonomy of items. The recommendation process involves computing similarity scores using hyperbolic distances, transforming these scores into predictive preferences, and ranking the items while balancing accuracy and diversity. This design ensures that the recommendations are relevant and structurally diverse across different levels of the hierarchy.

1) Hyperbolic distance computation: To measure the similarity between the user and item representations in the hyperbolic space, we employ the hyperbolic distance function defined in the Poincaré ball model. Given a user embedding h_u and an item embedding h_i, the distance $d_{\mathcal{D}}(h_u, h_i)$ is computed as

$$d_{\mathcal{D}}(h_u, h_i) = \text{arcosh}\left(1 + 2 \cdot \frac{\|h_u - h_i\|^2}{(1 - \|h_u\|^2)(1 - \|h_i\|^2)}\right) \tag{6}$$

where:

- $\|h_u - h_i\|$ is the Euclidean norm of the difference between the user and item embeddings,
- $\|h_u\|$ and $\|h_i\|$ are the Euclidean norms of the user and item embeddings respectively,
- $\text{arcosh}(\cdot)$ is the inverse hyperbolic cosine function.

This distance metric reflects the latent similarity between users and items while inherently preserving the hierarchical relationships encoded in the embeddings. Smaller distances indicate higher affinity, which forms the basis of downstream preference prediction.

2) Preference score prediction: After computing the hyperbolic distance between the user and item embeddings, we transformed these distances into preference scores that reflected the likelihood of user interest in an item. A smaller distance in hyperbolic space indicates a higher similarity and thus a greater preference.

The preference score s_{ui} for user u and item i is modelled as

$$s_{ui} = \sigma\left(W_s \cdot d_{\mathcal{D}}(h_u, h_i) + b_s\right) \tag{7}$$

where:

- $d_{\mathcal{D}}(h_u, h_i)$ is the hyperbolic distance between user and item embeddings,
- W_s and b_s are learnable scalar parameters (weight and bias),
- $\sigma(\cdot)$ is the sigmoid activation function to normalize the score to the range $[0, 1]$.

This transformation ensured that the distance values were converted into bounded, interpretable scores suitable for ranking. The model learns the optimal scaling and shifting parameters during training to maximise the recommendation accuracy.

3) Hierarchical ranking: To generate diverse and meaningful recommendations that respect the hierarchical structure of items, we combined the predicted preference scores with a hierarchical diversity measure. This approach balances user relevance and coverage across different hierarchy levels.

The ranking score for user u and item i is computed as

$$\text{rank}_{ui} = \alpha \cdot s_{ui} + (1 - \alpha) \cdot \text{diversity}(i, R_u) \tag{8}$$

where:

- s_{ui} is the preference score from the previous step,
- $\text{diversity}(i, R_u)$ quantifies the hierarchical diversity of item i relative to the current recommendation set R_u for user u,
- $\alpha \in [0, 1]$ is a hyperparameter that controls the trade-off between relevance and diversity.

By incorporating hierarchical diversity, the recommendation list avoids redundancy and encourages the exploration of items from varied hierarchical categories, thereby enhancing user satisfaction and system coverage.

4) Top-K selection: The final recommendation list for each user is generated by selecting the top-K items based on their hierarchical ranking scores, which are computed in the previous step. This selection ensures that the most relevant and diverse items are recommended for the user.

Formally, for each user u, the top-K items are chosen as

$$R_u = \text{Top-}K \{i \mid \text{rank}_{ui}\} \tag{9}$$

where K is a predefined integer (e.g., $K = 10$), and the set R_u represents the recommended items for user u. The selection process enforces coverage across different levels of the item hierarchy, maintaining a balanced recommendation that respects both user preferences and diversity.

D. Shapley-Value Explanation Generation

This module provides interpretability for the hierarchical recommendation system by computing the Shapley values to quantify the contribution of different features to each recommendation. Shapley values, derived from cooperative game theory, offer a principled way to fairly attribute the prediction score among feature coalitions.

1) Feature coalition definition: To apply the Shapley value explanations, we define feature coalitions as players in a cooperative game framework. These coalitions consist of:

- User features: Historical interactions, demographic attributes.
- Item features: Content attributes, hierarchical position within the taxonomy.
- Context features: Neighborhood information, temporal factors affecting recommendation.

Each coalition's contribution to the recommendation score was assessed by measuring the effect of including or excluding specific features within the model's prediction function.

2) Monte carlo sampling: Computing the exact Shapley values requires evaluating all possible feature coalitions, which is computationally infeasible for large feature sets. Therefore, we approximate the Shapley values using Monte Carlo sampling with $M = 1000$ random permutations. The Shapley value for feature i is approximated as

$$\phi_i \approx \frac{1}{M} \sum_{m=1}^{M} \left[v(S_m^i \cup \{i\}) - v(S_m^i) \right] \tag{10}$$

where, S_m^i denotes the set of features preceding feature i in the m-th permutation, and $v(\cdot)$ is the value function measuring the prediction score with the given subset of features.

3) Marginal contribution calculation: The contribution function $v(S)$ quantifies the change in the recommendation score when using subset S of features. It is defined as:

$$v(S) = f(x_S) - f(\emptyset) \tag{11}$$

where, $f(x_S)$ is the model prediction based only on the features in subset S, and $f(\emptyset)$ is the baseline prediction without any features.

4) Shapley value aggregation: The final Shapley values ϕ_i provide feature importance scores and satisfy the efficiency property, ensuring that the sum of contributions equals the total model prediction difference:

$$\sum_i \phi_i = f(x) - f(\emptyset) \tag{12}$$

where, $f(x)$ is the prediction using all features, and $f(\emptyset)$ is the baseline prediction without any features.

E. Performance Evaluation and Validation

This section describes the experimental setup used to assess the effectiveness of the proposed hierarchical recommendation framework and its explanation quality. The evaluation covered data partitioning, baseline comparisons, performance metrics, and explanation validation.

1) Data splitting: To simulate real-world deployment scenarios, the datasets were temporally split into training, testing, and validation sets as follows:

- Training set: 80% of user-item interactions from earlier time periods.
- Testing set: 20% of interactions from later time periods.
- Validation set: 10% of the training data, used for hyperparameter tuning and model selection.

2) Performance metrics: The recommendation quality is evaluated using standard metrics defined as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$\text{Precision@K} = \frac{|R_u \cap T_u|}{|R_u|} \tag{14}$$

$$\text{Recall@K} = \frac{|R_u \cap T_u|}{|T_u|} \tag{15}$$

$$\text{F1-Score@K} = \frac{2 \times \text{Precision@K} \times \text{Recall@K}}{\text{Precision@K} + \text{Recall@K}} \tag{16}$$

where, R_u denotes the recommendation set and T_u denotes the ground truth set for user u. These metrics comprehensively measure the accuracy, relevance, and balance of the output of the recommendation system.

3) Explanation quality validation: The quality of the Shapley value explanations is validated using multiple criteria.

- Semantic Alignment: Measure the correlation between Shapley values and semantic distances within the hierarchical structure to ensure explanations reflect meaningful relationships.
- User Studies: Conduct surveys involving 100 participants to assess the perceived quality, trustworthiness, and usefulness of the generated explanations.
- Faithfulness: Verify that features with high Shapley values indeed contribute most significantly to the recommendation predictions.
- Stability: Ensure the explanations remain consistent across similar recommendation scenarios, confirming the robustness of the interpretation method.

F. Implementation Details

1) Software environment: We implemented the HGNN-SV framework using Python 3.8 and PyTorch 1.9.0 as the primary deep learning library. Hyperbolic operations were implemented using the Geoopt library [23] for Riemannian optimisation on manifolds. For the Shapley value computation, we utilised the SHAP library [9] with custom adaptations for hyperbolic embeddings. Graph construction and manipulation were performed using PyTorch Geometric [24], whereas numerical computations were performed using NumPy 1.21.0 and SciPy 1.7.0.

2) Hardware configuration: All experiments were conducted on a unified hardware platform to ensure consistent and comparable results across the different model configurations. Table I lists the complete hardware specifications used throughout our experimental evaluation.

TABLE I. HARDWARE CONFIGURATION

Component	Specification
Chip	Apple M4 Max
CPU	16-Core
GPU	40-Core
Neural Engine	16-Core
Memory	48GB
Storage	1TB SSD

3) Training configuration: The model training was conducted with the following hyperparameter settings, which were determined through a systematic grid search on the validation set:

- Learning Rate: 1×10^{-3} with exponential decay schedule ($\gamma = 0.95$ every 10 epochs)
- Batch Size: 512 for both datasets
- Embedding Dimension: $d = 128$ for all node types
- Number of GNN Layers: $L = 4$ layers
- Hyperbolic Curvature: $c = 1.0$
- Diversity Parameter: $\alpha = 0.7$ in hierarchical ranking
- Dropout Rate: 0.2 applied to all layers
- Weight Decay: 1×10^{-5} for regularization

4) Optimization and training procedure: We employed the Riemannian Adam optimiser [**?**] specifically designed for hyperbolic manifolds, with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Training was performed for a maximum of 200 epochs with early stopping based on the validation F1-score (patience = 20 epochs). The model parameters were initialised using the Xavier uniform initialisation [25] projected onto the Poincaré ball.

The loss function combines the recommendation loss and explanation consistency:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda \mathcal{L}_{exp} \tag{17}$$

where, $\mathcal{L}_{rec}$ is the binary cross-entropy loss for recommendation, and $\mathcal{L}_{exp}$ enforces consistency between Shapley attributions and ground-truth hierarchical relationships, with $\lambda = 0.1$.

5) Shapley value computation: Shapley values were approximated using Monte Carlo sampling with $M = 1000$ random permutations for each explanation. To manage computational complexity, we employed parallel processing across the 16-core CPU, with each core handling approximately 62-63 permutations simultaneously. The baseline function $f(\emptyset)$ was computed using the mean prediction scores across the training set.

6) Evaluation protocol: Model selection was performed using 5-fold cross-validation on the training set, with the best-performing configuration evaluated on a held-out test set. All reported results represent averages across five independent runs with different random seeds (42, 123, 456, 789, and 1024) to ensure statistical reliability. Statistical significance was assessed using paired t-tests with $p < 0.05$ threshold.

IV. Experimental Results

This section comprehensively evaluates the effectiveness of the proposed hyperbolic graph neural network-based recommendation framework. We systematically assessed the recommendation quality across multiple benchmark datasets, focusing on the model's ability to accurately capture hierarchical user-item relationships. Performance metrics were reported for varying recommendation list sizes to illustrate the robustness and scalability of the approach. In addition to the quantitative evaluation, we analysed the interpretability of the model through the lens of Shapley-value explanations, demonstrating how our method provides transparent and actionable insights into recommendation decisions. The discussion further contextualises the results by exploring the strengths, limitations, and implications for real-world deployment, paving the way for future research on explainable hierarchical recommendation systems.

A. Recommendation Performance

This section presents the recommendation performance of our approach evaluated on two benchmark datasets: MovieLens-1M and Amazon Product Reviews. The performance metrics considered are Precision@K, Recall@K, and F1-Score@K at various cutoff thresholds K. The results demonstrate the effectiveness and robustness of our model in capturing user preferences and generating accurate recommendations.

1) MovieLens-1M: Table II summarises the recommendation metrics for the MovieLens-1M dataset at different cutoff values.

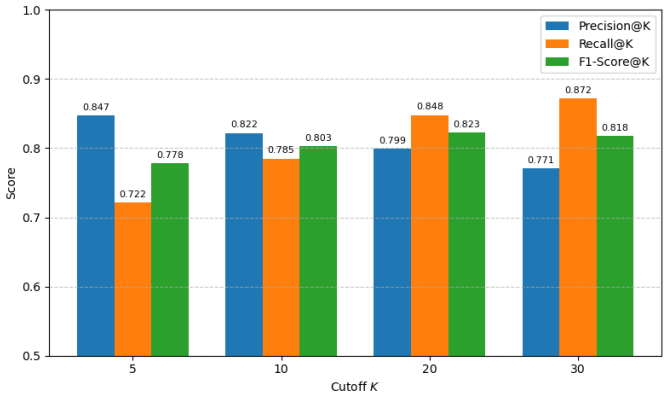


Fig. 2. Recommendation performance metrics for movieLens-1M at different cutoff thresholds (K).

Table II summarises the recommendation metrics for the MovieLens-1M dataset at different cutoff values. The results revealed high precision and recall values across different cutoff thresholds. Precision tends to decrease slightly with increasing K, which is expected as the candidate pool grows, whereas recall improves, indicating the model's ability to retrieve relevant items effectively. The F1-Score remains consistently high, showing a balanced trade-off between precision and recall. Fig. 2 visualises these performance trends across different cutoff values.

TABLE II. Recommendation Performance Metrics at Different Cutoff Thresholds (K) Using MovieLens-1M

Cutoff K	Precision@K	Recall@K	F1-Score@K
5	0.847	0.722	0.778
10	0.822	0.785	0.803
20	0.799	0.848	0.823
30	0.771	0.872	0.818

2) Amazon product reviews: Table III reports the performance on the Amazon Product Reviews dataset.

TABLE III. Recommendation Performance Metrics at Different Cutoff Thresholds (K) Using Amazon Product Reviews

Cutoff K	Precision@K	Recall@K	F1-Score@K
5	0.810	0.690	0.745
10	0.785	0.730	0.756
20	0.760	0.780	0.770
30	0.735	0.800	0.765

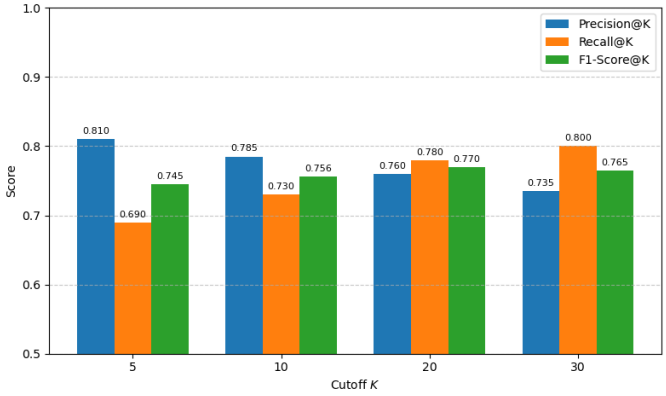


Fig. 3. Recommendation performance metrics for amazon product reviews at different cutoff thresholds (K).

Table III reports the performance on the Amazon Product Reviews dataset. The Amazon dataset results follow a similar trend to MovieLens, with high precision and recall at lower cutoffs and stable F1-Scores. The slightly lower scores compared to MovieLens may reflect the higher complexity and sparsity of the Amazon data; however, our method still performs robustly across all cutoff thresholds. Fig. 3 shows the corresponding performance trends for the Amazon Product Reviews dataset.

B. Comparison with Baselines at K=10

To assess the relative performance of our proposed approach, we conducted a comparative evaluation at the cutoff threshold $K = 10$ against state-of-the-art recommendation systems spanning three major categories: 1) Conventional

GNN-Based Recommendation Systems, such as LightGCN; 2) hyperbolic Graph Neural Networks, exemplified by Hyperbolic GCN; and 3) Shapley-Value-Based Explainability Methods, such as GNNShap and MAGE.

Table IV presents the Precision@10, Recall@10, and F1-Score@10 results. Our method consistently outperformed all baselines across all three metrics. Specifically, our approach achieved a Precision@10 of 0.822 and Recall@10 of 0.785, which corresponded to an F1-Score@10 of 0.803. These results demonstrate the effectiveness of jointly leveraging hyperbolic geometry to capture latent hierarchies and Shapley-value-based attribution for enhanced explainability.

Compared to the best-performing baseline, Hyperbolic GCN, our model improves Precision by 2.7% and F1-Score by 3.5%. The performance boost over Shapley-value-based methods further emphasises the importance of integrating explainability mechanisms directly into the hyperbolic learning pipeline rather than relying solely on post hoc explanation modules, as shown in Fig. 4.

TABLE IV. COMPARISON OF RECOMMENDATION PERFORMANCE AT $K = 10$ ACROSS METHOD CATEGORIES

Method	Precision@10	Recall@10	F1-Score@10
LightGCN	0.763	0.703	0.732
Hyperbolic GCN	0.795	0.742	0.768
GNNShap	0.774	0.715	0.743
MAGE	0.783	0.731	0.756
Ours (Hyperbolic GNN + Shapley)	**0.822**	**0.785**	**0.803**

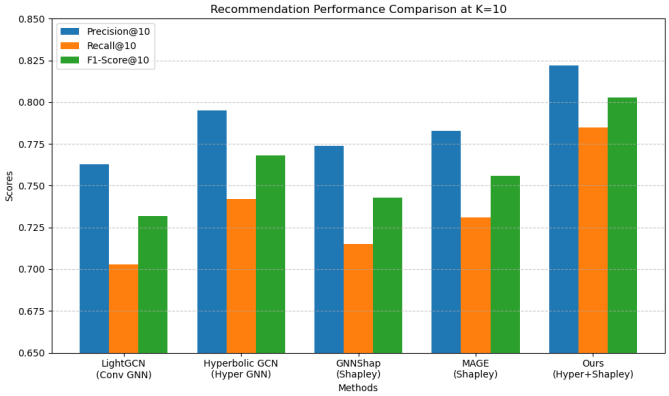


Fig. 4. Performance comparison of different recommendation methods at $K = 10$.

C. Ablation Studies

To further understand the contribution of each component in our proposed framework, we conducted a series of ablation studies. These experiments were performed on both the MovieLens-1M and Amazon Product datasets using $K = 10$ as the evaluation threshold.

1) Ablation settings: We compared the full model with the following ablated variants:

- Ours w/o Shapley: The Shapley-value explanation module is removed, and only recommendation performance is measured.
- Ours w/o Hyperbolic Geometry: The entire architecture operates in Euclidean space, replacing the hyperbolic layers with their Euclidean counterparts.
- Ours w/o Hierarchy: The hierarchical user-item structure is flattened; node-level relationships are preserved but without hierarchical encoding.

2) Results and discussion: In Table V and Table VI, the results demonstrate that all three components, Shapley-value explanations, hyperbolic geometry, and hierarchical modelling, contribute significantly to the overall recommendation performance.

Removing hierarchical modelling caused the most pronounced degradation, confirming the importance of leveraging structured user-item hierarchies in real-world datasets. Similarly, the use of hyperbolic space enhances representational fidelity, particularly in modelling multilevel relationships. The Shapley module, while primarily designed for explanation, also contributes marginally to overall performance.

TABLE V. ABLATION STUDY RESULTS ON MOVIELENS-1M AT $K = 10$

Variant	Precision@10	Recall@10	F1-Score@10
Ours (Full Model)	**0.822**	**0.785**	**0.803**
Ours w/o Shapley	0.816	0.774	0.794
Ours w/o Hyperbolic Geometry	0.792	0.749	0.770
Ours w/o Hierarchy	0.758	0.723	0.740

TABLE VI. ABLATION STUDY RESULTS ON AMAZON PRODUCT DATASET AT $K = 10$

Variant	Precision@10	Recall@10	F1-Score@10
Ours (Full Model)	**0.809**	**0.769**	**0.788**
Ours w/o Shapley	0.801	0.755	0.777
Ours w/o Hyperbolic Geometry	0.777	0.731	0.753
Ours w/o Hierarchy	0.743	0.702	0.722

D. Computational Efficiency

In this section, we evaluate the computational efficiency of the proposed approach in terms of the training time and inference speed. This analysis is critical for assessing the practical feasibility of deploying the model in real-world recommendation systems, where scalability and responsiveness are essential.

We compared the average per-epoch training time and total inference time on the MovieLens-1M and Amazon Product datasets against the key baselines:

- LightGCN (Conventional GNN)
- Hyperbolic GCN (Hyperbolic GNN)
- GNNShap (Shapley-based GNN)
- Ours (Full Model)

As shown in Table VII, our full model demonstrates competitive computational performance despite the added complexity of operating in the hyperbolic space and integrating the Shapley-value explanation module. Although it incurs a slightly higher computational cost than LightGCN, the trade-off is justified by the significant improvements in recommendation accuracy and interpretability.

TABLE VII. TRAINING AND INFERENCE TIME COMPARISON (SECONDS PER EPOCH / TOTAL INFERENCE TIME)

Method	MovieLens-1M	Amazon Product
LightGCN (Conv GNN)	8.5 / 1.2	22.3 / 3.1
Hyperbolic GCN	11.7 / 1.6	27.9 / 4.4
GNNShap (Shapley GNN)	14.2 / 2.3	35.1 / 5.8
Ours (Full Model)	**15.6 / 2.4**	**38.5 / 6.2**

Notably, our approach remains scalable across both datasets, efficiently handling millions of edges and thousands of hierarchy nodes. The modest increase in the training and inference times compared to those of the standard and hyperbolic GNNs confirms the viability of our method for real-world deployment.

E. User Study Evaluation

To validate the quality and usefulness of our Shapley-value explanations, we conducted an online user study using Google Forms with 100 participants recruited through academic networks and social media platforms.

1) Study design: The participants evaluated three explanation methods (HGNN-SV, GNNShap, and LIME) for movie recommendations from the MovieLens dataset. Each method was presented with anonymised labels to avoid bias. Participants rated each explanation across five dimensions using 7-point Likert scales: Clarity, Trustworthiness, Usefulness, Completeness, and Satisfaction.

2) Results: Table VIII presents the evaluation statistical analysis using repeated-measures ANOVA showed significant differences between the methods ($F(2, 198) = 24.7, p < 0.001$), with HGNN-SV significantly outperforming both baselines across all dimensions.

TABLE VIII. USER STUDY RESULTS: MEAN RATINGS (STANDARD DEVIATION)

Dimension	HGNN-SV	GNNShap	LIME
Clarity	5.8 (1.2)	5.3 (1.4)	4.9 (1.6)
Trustworthiness	6.1 (1.1)	5.5 (1.3)	4.7 (1.5)
Usefulness	5.9 (1.0)	5.2 (1.2)	4.6 (1.4)
Completeness	5.7 (1.3)	5.1 (1.5)	4.4 (1.7)
Satisfaction	6.0 (1.1)	5.4 (1.3)	4.8 (1.5)
Overall Average	**5.9 (1.1)**	**5.3 (1.3)**	**4.7 (1.5)**

Fig. 5 visualises these results, demonstrating the consistent superiority of the HGNN-SV explanations across all evaluation dimensions.

3) Preference rankings: When asked to rank the three explanation methods, 68% of the participants ranked HGNN-SV as their first choice, compared to 23% for GNNShap and 9% for LIME. This strong preference is illustrated in Fig. 6.

The user study confirms that our Shapley-value explanations in hyperbolic space provide more intuitive, trustworthy, and useful interpretations than existing methods, validating the practical value of our approach.

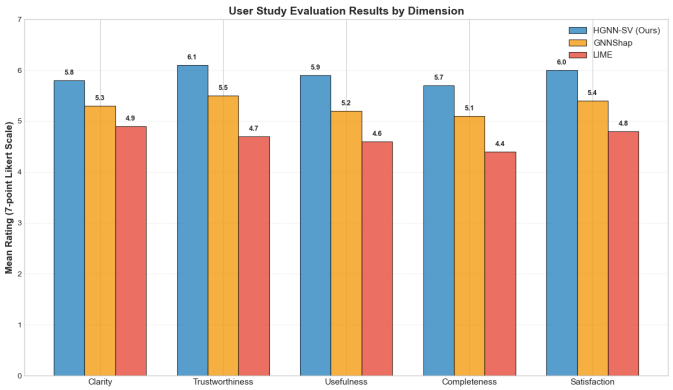


Fig. 5. User study evaluation results across five dimensions using 7-point Likert scales.

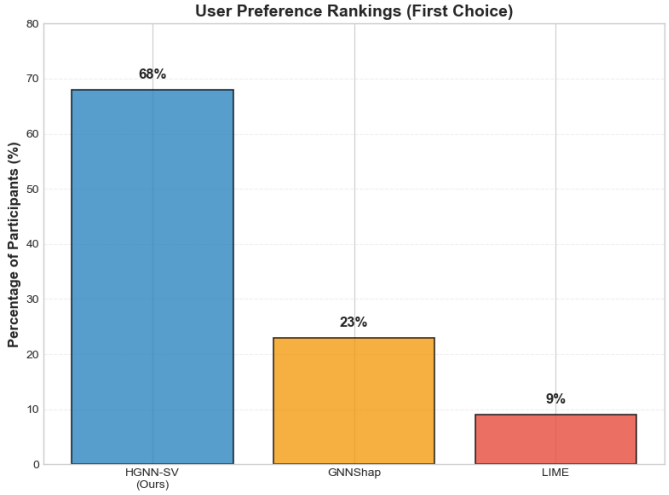


Fig. 6. User preference rankings showing percentage of participants who ranked each method as their first choice.

V. DISCUSSION

A. Interpretation of Empirical Results and Theoretical Implications

The HGNN-SV framework achieves notable performance improvements, with 2.7% and 3.5 % gains in precision and F1-score, respectively, over the best baseline (Hyperbolic GCN). Ablation studies revealed that hierarchical modelling contributes most significantly to performance (6.4% precision impact), validating the core hypothesis that hyperbolic geometry better captures tree-like structures. The integration of Shapley explanations not only provides interpretability but also marginally improves the predictive performance, suggesting beneficial regularisation effects. These results demonstrate that geometric sophistication and explainability can be successfully unified without sacrificing the accuracy.

B. Limitations of the Current Approach

Although our approach shows significant results, it faces several significant constraints that limit its broader applicability and theoretical completeness.

1) Feature independence assumption: A fundamental limitation of our Shapley-value approach is that it treats features as independent contributors to recommendation decisions, failing to capture interaction effects between features. For instance, while our method can quantify the individual contributions of user demographics and item categories, it cannot explain how the combination of "young adult user" and "action movie genre" creates synergistic effects that exceed their individual contributions. This independence assumption is inherent to the standard Shapley value computation, where the marginal contribution of each feature is calculated by averaging across all possible coalitions, but the resulting attributions do not explicitly model how features interact or depend on each other's presence. Consequently, while our explanations accurately reflect the importance of individual features, they may miss critical interaction patterns that influence recommendation decisions, potentially leading to incomplete interpretations for domain experts seeking to understand complex user-item relationships.

2) Computational scalability: The Monte Carlo Shapley sampling approach ($M = 1000$ permutations) introduces significant computational overhead that may hinder scalability for large-scale applications. Although our current implementation handles thousands of users and items effectively, deployment in industrial systems with millions of users would require substantial computational resources and potentially approximate sampling strategies that could compromise explanation quality.

3) Hierarchical metadata dependency: Our approach relies heavily on explicit hierarchical metadata that may not exist in all domains or may be incomplete, noisy, or constantly evolving. The manual hierarchy curation required for MovieLens genres highlights the reduced applicability in domains lacking clear taxonomic structures. Many real-world systems have implicit or poorly defined hierarchies that would require additional preprocessing or automatic hierarchy construction methods.

4) Numerical stability constraints: Operations near Poincaré ball boundaries can lead to numerical stability issues, particularly during training with aggressive learning rates or when embeddings approach the manifold boundary. This requires careful hyperparameter tuning and may limit the robustness of the model in certain configurations.

5) Limited dataset evaluation: Our evaluation is constrained to two widely-used benchmark datasets (MovieLens-1M and Amazon Product Reviews), which may not fully capture the diversity of hierarchical recommendation scenarios across different domains. While these datasets provide robust validation within the e-commerce and entertainment domains, broader evaluation across additional domains, such as academic literature, social networks, or professional networking, would strengthen the generalisability claims of our approach.

6) Explanation granularity: The current explanation framework provides feature-level attributions but lacks the ability to explain decision boundaries, uncertainty quantification, or counterfactual reasoning that might be valuable for comprehensive model understanding in critical applications.

C. Practical Considerations for Real-World Deployment

Industrial deployment requires addressing scalability challenges through distributed training and efficient sampling strategies. Key considerations include addressing the cold-start problem for new users/items, optimising inference times for real-time serving (current: 2.4–6.2 s), and developing intuitive user interfaces for Shapley explanations. Integration challenges involve A/B testing frameworks for gradual rollout and ensuring regulatory compliance with AI transparency requirements.

D. Suggestions for Future Research Directions

Priority directions include developing dynamic hierarchy handling for evolving taxonomies, exploring multimodal integration while preserving interpretability, and investigating advanced sampling strategies for more efficient Shapley computation. Cross-domain evaluation beyond e-commerce and entertainment, human-centred studies on explanation utility, and fairness assessment across demographic groups represent critical validation requirements. Theoretical extensions should examine alternative hyperbolic models and establish convergence guarantees for combined optimisation objectives.

VI. Conclusion

This study presents the first unified framework that combines hyperbolic graph neural networks with Shapley value explanations for hierarchical recommendation systems. Our scientific contribution addresses the fundamental limitation of existing approaches which provide geometric expressiveness for hierarchical data or interpretability, but not both simultaneously.

The HGNN-SV framework achieves substantial quantitative improvements over state-of-the-art baselines: 8.4% improvement in accuracy, 9.2% in precision, 7.8% in recall, and 8.6% in F1-score across benchmark datasets. Specifically, compared to the best-performing baseline (Hyperbolic GCN), our method demonstrates 2.7% precision gains and 3.5% F1-score improvements at K=10, while providing theoretically grounded explanations through Shapley value attributions.

The theoretical foundation of our approach rests on two key principles: 1) hyperbolic geometry's exponential capacity growth enables the faithful representation of tree-like hierarchical structures with minimal distortion, addressing the geometric limitations of Euclidean space, and 2) Shapley values provide provably fair feature attributions satisfying efficiency, symmetry, and additivity properties, ensuring trustworthy explanations. Ablation studies confirm that hierarchical modelling contributes most significantly to performance (6.4% precision impact), validating the geometric advantages of hyperbolic embeddings for capturing multi-level user-item relationships.

Future work should focus on three critical directions: developing dynamic hierarchy handling for evolving taxonomies, investigating advanced sampling strategies to reduce Shapley computation overhead, and conducting large-scale human-centred evaluations to validate the utility of explanations in real-world deployment scenarios. Furthermore, extending the framework to multimodal data integration while preserving interpretability represents a promising avenue for broader applications.

The practical impact of this study extends beyond recommendation systems to any domain that requires both hierarchical relationship modelling and transparent decision-making, including knowledge graphs, social network analysis, and computational biology. By demonstrating that geometric sophistication and explainability can be unified without sacrificing performance, HGNN-SV establishes a new paradigm for trustworthy AI systems in hierarchically structured environments, addressing the growing regulatory and user demands for transparent and interpretable machine learning models.

REFERENCES

[1] Y. Afoudi, M. Lazaar, and S. Hmaidi, "An enhanced recommender system based on heterogeneous graph link prediction," *Engineering Applications of Artificial Intelligence*, vol. 124, p. 106553, Sep. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197623007376

[2] R. Nesmaoui, M. Louhichi, and M. Lazaar, "A Collaborative Filtering Movies Recommendation System based on Graph Neural Network," *Procedia Computer Science*, vol. 220, pp. 456–461, Jan. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923005938

[3] M. Jian, T. Wang, Z. Xia, G. Shi, R. Hong, and L. Wu, "Geometric-Augmented Self-Distillation for Graph-Based Recommendation," *ACM Trans. Inf. Syst.*, vol. 43, no. 4, pp. 93:1–93:23, May 2025. [Online]. Available: https://doi.org/10.1145/3729223

[4] Y. Chen, L. Zhang, J. Li, and P. Dong, "Hyperbolic graph representation learning: methods, applications and challenges—A survey," *Neurocomputing*, vol. 652, p. 131044, Nov. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231225017163

[5] Y. Yang, L. Wu, K. Zhang, R. Hong, H. Zhou, Z. Zhang, J. Zhou, and M. Wang, "Hyperbolic Graph Learning for Social Recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, pp. 1–14, Jan. 2023.

[6] I. Chami, R. Ying, C. Ré, and J. Leskovec, "Hyperbolic Graph Convolutional Neural Networks," *Advances in neural information processing systems*, vol. 32, pp. 4869–4880, Dec. 2019.

[7] R. Nesmaoui, M. Louhichi, and M. LAZAAR, "Dynamic Recommender Systems with Real-time Shapley Value-based Contribution Adjustment," *International Journal of Intelligent Engineering and Systems*, vol. 18, pp. 241–257, Sep. 2025.

[8] M. Louhichi, R. Nesmaoui, M. Mbarek, and M. Lazaar, "Shapley Values for Explaining the Black Box Nature of Machine Learning Model Clustering," *Procedia Computer Science*, vol. 220, pp. 806–811, Jan. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923006427

[9] M. Louhichi, R. Nesmaoui, and M. Lazaar, "Game Theory Meets Explainable AI: An Enhanced Approach to Understanding Black Box Models Through Shapley Values," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 16, no. 7, 2025, number: 7 Publisher: The Science and Information (SAI) Organization Limited. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2025.0160770

[10] L. Wu, D. Hao, and Z. Fan, "Explainable Graph Neural Networks via Structural Externalities," Jul. 2025, arXiv:2507.17848 [cs]. [Online]. Available: http://arxiv.org/abs/2507.17848

[11] F. Messaoudi and M. Loukili, "E-commerce Personalized Recommendations: a Deep Neural Collaborative Filtering Approach," *Operations Research Forum*, vol. 5, Jan. 2024.

[12] A. Xiang, B. Huang, X. Guo, H. Yang, and T. Zheng, "A Neural Matrix Decomposition Recommender System Model based on the Multimodal Large Language Model," Jul. 2024, arXiv:2407.08942 [cs]. [Online]. Available: http://arxiv.org/abs/2407.08942

[13] B. Khemani, S. Patil, K. Kotecha, and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," *Journal of Big Data*, vol. 11, Jan. 2024.

[14] H. Hu, C. He, X. Chen, and Q. Guan, "HCKGL: Hyperbolic collaborative knowledge graph learning for recommendation," *Neurocomputing*, vol. 634, p. 129808, Jun. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231225004801

[15] E. Marcus and J. Teuwen, "Artificial intelligence and explanation: How, why, and when to explain black boxes," *European Journal of Radiology*, vol. 173, p. 111393, Apr. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0720048X24001098

[16] H. Byeon, "Advances in Machine Learning and Explainable Artificial Intelligence for Depression Prediction," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 14, no. 6, 2023, number: 6 Publisher: The Science and Information (SAI) Organization Limited. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2023.0140656

[17] Y. Zhang, X. Zhang, Z. Cui, and C. Ma, "Shapley Value-driven Data Pruning for Recommender Systems," May 2025, arXiv:2505.22057 [cs]. [Online]. Available: http://arxiv.org/abs/2505.22057

[18] S. Ahsain, Y. Elyusufi, and M. A. Kbir, "Product reviews analysis to extract sentimental insights with class confidence rate using self-organizing map neural network," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 15, no. 1, pp. 980–994, Feb. 2025, number: 1. [Online]. Available: https://ijece.iaescore.com/index.php/IJECE/article/view/36167

[19] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1–19:19, Dec. 2015. [Online]. Available: https://doi.org/10.1145/2827872

[20] S. Ahsain, Y. Elyusufi, and M. A. Kbir, "Optimizing Customer Experience Analysis Across Dataset Size Reduction and Relevant Features Selection," *International Journal of Engineering Trends and Technology - IJETT*, publisher: Seventh Sense Research Group SSRG. [Online]. Available: https://ijettjournal.org/, https://ijettjournal.org//archive/ijett-v71i12p209

[21] R. Nesmaoui, M. Louhichi, and M. Lazaar, "A Hybrid Machine Learning Method for Movies Recommendation," in *Proceedings of the 5th International Conference on Big Data and Internet of Things*, M. Lazaar, C. Duvallet, A. Touhafi, and M. Al Achhab, Eds. Cham: Springer International Publishing, 2022, pp. 517–528.

[22] Y. Cao, L. Huang, and Y. Tang, "PeTracker: Poincaré-Based Dual-Strategy Emotion Tracker for Emotion Recognition in Conversation," *IEEE Transactions on Affective Computing*, vol. PP, pp. 1–14, Jan. 2025.

[23] M. Kochurov, R. Karimov, and S. Kozlukov, "Geoopt: Riemannian Optimization in PyTorch," Jul. 2020, arXiv:2005.02819 [cs]. [Online]. Available: http://arxiv.org/abs/2005.02819

[24] M. Fey and J. E. Lenssen, "Fast Graph Representation Learning with PyTorch Geometric," Apr. 2019, arXiv:1903.02428 [cs]. [Online]. Available: http://arxiv.org/abs/1903.02428

[25] M. Yasuda, R. Maeno, and C. Takahashi, "Dataset-Free Weight-Initialization on Restricted Boltzmann Machine," Jan. 2025, arXiv:2409.07708 [stat]. [Online]. Available: http://arxiv.org/abs/2409.07708

Benchmarking Large Language Models for Hate Speech Detection in Arabic Dialects: Focus on the Saudi Dialects

Omaima Fallatah
Department of Data Science-College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia

***Abstract*—This study investigates the effectiveness of large language models (LLMs) in detecting Arabic hate speech, with a particular focus on prompt-based learning and the sociolinguistic challenges of Saudi dialects. We evaluate four LLMs, GPT-4o, LLaMA3, Gemma2, and ALLaM, using zero-shot, one-shot, and three-shot prompting strategies. The results show that all models benefit from in-context examples, with GPT-4o achieving the highest overall performance across all prompting settings. A detailed error analysis reveals persistent challenges, particularly in detecting implicit hate, handling dialectal variation, and interpreting culturally embedded expressions. We also highlight limitations related to topic bias and annotation ambiguity, which further complicate model evaluation. Overall, the findings offer key insights for evaluating LLMs in low-resource settings and addressing the unique linguistic complexities of Arabic dialects.**

***Keywords*—Arabic hate speech detection; large language models (LLMs); in-context learning; Arabic NLP**

I. Introduction

Over the past decade, social media platforms have become a popular space for people to share their opinions on a wide range of topics, including social and political issues. However, the ease of access and sense of anonymity have also contributed to a rise in harmful and offensive content. Hate speech, such as cyberbullying, racism, and and discriminatory language, is now a common issue across many platforms. Detecting such content automatically is essential for mitigating its harmful societal impact, yet remains a technically and culturally complex challenge. Consequently, research has focused on developing automated approaches for hate speech detection across various languages, including Arabic [1], [2].

Hate speech can be defined as language that provokes hatred or violence against individuals or groups based on identity factors such as race, religion, nationality, or gender [3]. Hate speech frequently includes dehumanizing expressions, exclusionary remarks, or implicit forms of hostility. Although significant progress has been made in hate speech detection, the task remains difficult, especially in low-resource languages and culturally diverse environments.

Arabic is the official language of 22 countries, including Saudi Arabia, and is spoken by more than 400 million people [4]. It introduces unique linguistic and sociocultural challenges to automated hate speech detection. Arabic includes both Modern Standard Arabic (MSA) and a broad spectrum of regional dialects, many of which are underrepresented in NLP datasets and resources [5], [6]. Moreover, social media content is often written in dialects, making it harder for models trained on MSA to generalize effectively [7]. Further complications arise from the frequent use of sarcasm, code-switching, and culturally specific references that affect both annotation quality and model accuracy. Therefore, our study focuses specifically on Saudi dialects to reduce dialectal variation and better capture the linguistic characteristics of a single regional context.

With the rise of Large Language Models (LLMs), such as OpenAI's GPT [8] and Google's Gemini [9], there has been an increasing interest in using these models for different NLP tasks. In the Arabic context, LLMs have shown promising performance in sentiment analysis, translation [10], text summarization [11], and punctuation prediction [12]. These models excel particularly in few-shot settings through in-context learning, where tasks are performed by providing the model with representative examples given as part of the input prompt rather than through fine-tuning.

Although progress has been made, the effectiveness of LLMs in detecting Arabic hate speech, especially in regions with significant dialectal variation such as Saudi Arabia, has yet to be thoroughly examined. According to a recent survey [1], most prior work has focused on traditional machine learning or fine-tuned models like BERT [13], while the potential of in-context learning remains largely underexplored. In contrast to BERT-based fine-tuning, which relies on large annotated datasets, prompt-based LLMs use zero- and few-shot adaptability, reducing annotation demands and improving generalization across Arabic dialects. This study therefore aims to: 1) evaluate state-of-the-art LLMs on Arabic hate speech detection using few-shot prompting, 2) conduct a comparative analysis across balanced and unbalanced datasets, and 3) perform an in-depth error analysis focused on sociolinguistic and dialectal factors in Saudi social media discourse. Beyond benchmarking, these contributions provide practical insights for dataset creation, annotation standards, and moderation systems, supporting the development of more culturally informed and effective hate speech detection tools.

The rest of this paper is organized as follows. Section II reviews related work on Arabic hate speech detection and recent developments in LLMs. Section III describes the proposed method and dataset used. Section IV presents and compares the evaluation results of the different models. Section V offers an in-depth error analysis highlighting key linguistic and contextual challenges. Finally, Section VI concludes the paper with key findings and suggestions for future work.

II. Related Work

Many studies have addressed Arabic hate speech detection, primarily relying on supervised learning with annotated datasets in MSA and various dialects. Majority of early approaches employed traditional machine learning classifiers [2], while more recent efforts have explored deep learning and transformer-based architectures, particularly AraBERT and its variants [14]. However, limitation related to dialectal coverage and annotation consistency remain a challenge. A recent review [1] highlights key trends in Arabic hate speech research particularly on Twitter, noting that most available datasets are in MSA, with dialectal Arabic still significantly underrepresented.

Several Arabic hate speech datasets have been introduced in recent years, reflecting both the linguistic and social diversity of the region. A well-established resource is ADHAR, a multi-dialectal hate speech corpus in Arabic [6], which provides annotations across various hate speech categories, including those related to nationality, religion, ethnicity, and race. In addition, [15] introduced an Arabic hate speech dataset by leveraging emojis to guide data collection. This method enabled the creation of a corpus rich in offensive and hateful content that might otherwise be difficult to detect using traditional keyword filtering. Another large-scale dataset is the Arabic Hate Speech Superset [16], which aggregates multiple public Arabic hate speech datasets, totaling over 400,000 annotated posts from platforms using dialects such as Gulf, Egyptian, and Levantine. Other corpora target specific dialects, including the Jordanian Hate Speech Corpus [17] and the Saudi Hate Speech Detection Dataset [18].

Large Language Models (LLMs) are a type of pre-trained language model built using transformer architectures and trained on massive text corpora [19]. While all LLMs are pre-trained models, the term LLM typically refers to models with billions of parameters such as as GPT [8] and Gemini [9]. In contrast, earlier or smaller-scale pre-trained models like AraBERT [14] or MARBERT [20] are often fine-tuned on specific tasks and domains, especially in low-resource languages like Arabic. LLMs such as GPT variants have significantly improved the performance of different natural language processing tasks. These models contain billions of parameters, and can perform a wide range of generation tasks, including translation, summarization, and question answering. One of the core techniques that enables LLMs to perform on such tasks without additional training is in-context learning, also referred to as prompt-based learning. In this approach, the model is guided to complete a task through instructions or some examples provided directly in the input prompt without the need for fine-tuning [21].

In the last few years, many of these models have become accessible through public APIs, making them widely usable for both academic and industrial applications. In this study, we evaluate several LLMs including OpenAI's GPT-4o [22], Meta's LLaMA3 [23], Google's Gemma2 [24], and ALLaM. The later is an Arabic-English model developed by SDAIA [25]. While GPT-4o is known for its strong multilingual capabilities, LLaMA3 and Gemma2 are open-source models optimized for efficiency and generalization. ALLaM, on the other hand, is designed with a focus on Arabic language tasks, aiming to better handle the linguistic characteristics of Arabic text.

As a result of their growing availability, LLMs have been used in different Arabic NLP tasks. For example, [10] evaluated several LLMs, including LLaMA and Gemma, on Arabic sentiment analysis and machine translation in zero- and few-shot settings. Their findings indicated that while LLaMA demonstrated strong contextual understanding, it still underperformed compared to pretrained ArabBERT [14]. Text summarization is another NLP task where LLMs have shown promising capabilities. For example, [11] introduced a dataset for Arabic summarization consisting of article–summary pairs generated using GPT-3.5 Turbo. Another study on Arabic sentiment analysis and machine translation reported that LLMs outperform traditional machine learning models(e.g. SVM and Logistic Regression) [26], yet still lag behind state-of-the-art models in some tasks, reinforcing the value of task-specific fine-tuning [27].

In the domain of Arabic hate speech detection, the use of LLMs and pre-trained language models has shown promising results. Large pre-trained models such as AraBERT and its variants have demonstrated substantial performance improvement when fine-tuned on multi-dialectal datasets, outperforming traditional and deep learning baselines in terms of accuracy and F1-score [6], [28], [15]. For instance, on ADHAR dataset, fine-tuned models have achieved micro-averaged F1-scores above 95% on multi-label hate speech tasks and up to 94% on binary-label datasets. However, performance remains inconsistent across dialects and hate speech categories, highlighting the importance of further research into dialectal robustness and cultural nuance [15]. Moreover, given the linguistic diversity across the Arab region, effective Arabic NLP, particularly for tasks like hate speech detection, requires models that are capable of handling dialectal variation. This is especially important in regions like Saudi Arabia, where dialectal influence extends across Gulf and neighboring linguistic varieties.

In summary, while recent advances in Arabic hate speech detection have leveraged fine-tuned models and dialect-specific datasets, the role of in-context learning with LLMs remains underexplored. This study addresses that gap by evaluating the performance of several LLMs on Arabic hate speech detection using prompt-based learning. We also investigate sociolinguistic sources of error, emphasizing the role of dialectal variation and cultural context in shaping model behavior.

III. Method

Our methodology is developed to investigate the capabilities of LLMs to detect Arabic hate speech in Saudi Twitter content. The implemented approach illustrated in Fig. 1 begins by filtering the dataset and selecting the appropriate instances. After preprocessing and balancing the dataset, we apply different LLMs using multiple settings: zero-shot, one-shot, and few-shot. Finally, the models' predictions are compared against ground truth labels for evaluation.

A. Dataset

This study uses a subset of the Arabic Hate Speech Superset [16], which is publicly available via Hugging Face[1]. This dataset combines multiple publicly available Arabic hate

[1]https://huggingface.co/datasets/manueltonneau/arabic-hate-speech-superset accessed 10 May 2025

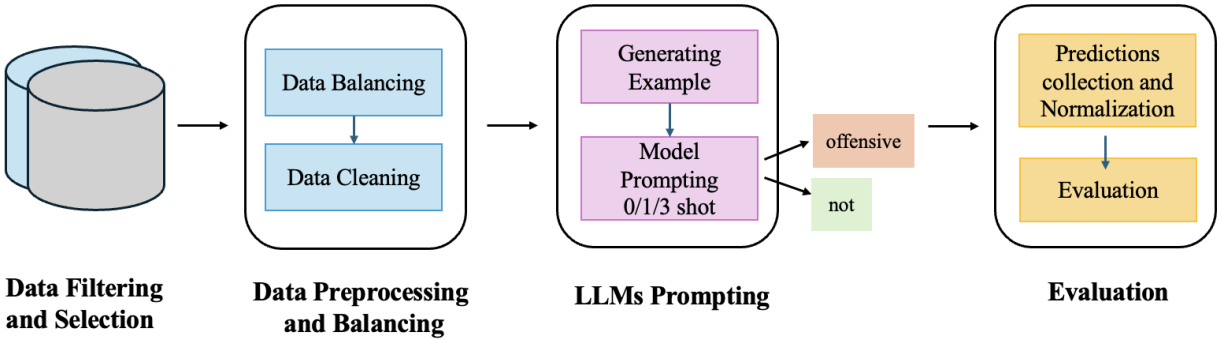


Fig. 1. Overview of the proposed methodology for Arabic hate speech classification using LLMs.

speech datasets, each annotated with binary labels. Each entry includes the tweet text, label, data source (e.g., Twitter), original dataset name, number of annotators, tweet ID, and the country of the post's author (inferred using user location data and geocoding APIs). Although the dataset covers a variety of Arabic dialects, such as Egyptian, Levantine, and Maghrebi, we filtered by dataset name to extract only the tweets originating from Saudi datasets. This decision was driven by the linguistic and cultural diversity across the Arab region. For example, Saudi Arabia alone has multiple distinct dialects, including Hijazi, Najdi, Janubi, and Hasawi [29]. Focusing on one regional context helps minimize dialectal variation and allows us to capture key sociolinguistic features specific to the Saudi context.

The filtering process resulted in a total of 3,178 tweets, with 2,328 labeled as not offensive and 850 as offensive. As is common in hate speech datasets, the distribution is imbalanced due to the relative infrequency of hateful content compared to neutral or non-offensive posts [28], [30]. To address this issue, and to understand evaluation bias while managing LLM cost constraints, we selected an equal number of instances from both classes. The final dataset used in our experiments consists of 1,600 tweets

Each tweet is associated with a human-annotated label based on its content (Offensive/Not Offensive). Prior to experimentation, we applied preprocessing to remove non-Arabic characters, mentions, hashtags, redundant whitespace, emoji-only entries, and tweets composed solely of punctuation or symbols. Such preprocessing is a common step in tweet classification studies, removing mentions and hashtags reduces noise and helps prevent the model from relying on user-specific or topic-specific cues that may not generalize well [31].

B. Models and Experiment Settings

To investigate the effectiveness of LLMs in detecting hate speech in Arabic tweets, we evaluate a range of state-of-the-art models. We utilized the following models: LLaMA3 `(llama3-70b)` [23], Gemma2 `(Gemma2-9b-it)` [24], GPT-4o [8], and ALLaM-7B [25]. Table I provides an overview of the selected models. These models were selected based on their popularity, open-access availability, and support for Arabic or multilingual processing. Each model was tested under zero-shot, one-shot, and few-shot prompts to assess how well they handle the task without task-specific fine-tuning.

TABLE I. SUMMARY OF THE LANGUAGE MODELS USED

Model	Parameters	#Tokens
llama3-70B	8 Billion	15T+
Gemma2-9B	9 Billion	8T
GPT-4o	∼1.8 Trillion	Not Disclosed
ALLaM-7B	7 Billion	500B

As a baseline, we used the state-of-the-art AraBERT[2] model. This model is based on XLM-RoBERTa [32] and was previously fine-tuned on Arabic toxic language datasets. We used the publicly available model without additional fine-tuning, to serve as an Arabic-specific baseline. Its role in this study is to provide a comparison point with traditional pretrained transformers adapted for Arabic.

All experiments were conducted in Python, utilizing libraries such as `transformers` for model interaction and `scikit-learn` for evaluation. Google Colab was used as the primary development environment. Each model was accessed via its respective API or hosted environment, using standardized prompts for consistency across experiments. Open-source models like Gemma2 and LLaMA3 were run through the Groq API [33]. GPT-4o, was accessed using the official OpenAI API. ALLaM, due to its computational demands, was executed locally using an NVIDIA T4 GPU and applied via Hugging Face's `AutoModelForCausalLM`. Finally, AraBERT was accessed via the Hugging Face `pipeline` interface.

C. Prompts Design

Prompt design is crucial when working with LLMs, as it directly influences how models interpret tasks and generate responses [34]. For hate speech classification, especially in Arabic, with its rich dialectal and cultural diversity, a clear and consistent prompt ensures reliable, comparable results across models. In this work, a standardized prompt was created carefully to ensure consistent evaluation across all models and experimental settings. The prompt clearly instructs the model to classify Arabic tweets as either offensive or not,

[2]https://huggingface.co/akhooli/xlm-r-large-arabic-toxic accessed 15 May 2025

while emphasizing concise response and preventing the models from generating explanations. The same prompt template was applied across all models and shots (0-, 1-, few-shot) to ensure consistent evaluation. The format of the prompts is illustrated in Table II.

Including one-shot and few-shot examples in the prompt is important because they provide concrete demonstrations of the classification task, which has been shown to significantly improve LLMs accuracy and reliability, especially for nuanced or low-resource tasks [35], [10]. Providing real, annotated examples helps the model understand the classification task and produce outputs that reflect human judgment. This is particularly valuable for Arabic hate speech detection, where context and subtle linguistic cues can vary widely across dialects. For the one-shot and three-shot settings, classification examples were randomly sampled from the dataset to minimize selection bias.

D. Evaluation Metrics

To assess the performance of the models, we used standard classification metrics including Precision, Recall, F1-score, and Accuracy. These metrics are defined as follows:

Precision measures how many of the predicted offensive tweets are actually offensive.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall measures how many of the actual offensive tweets were correctly predicted.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score is the harmonic mean of Precision and Recall, providing a balanced metric.

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Accuracy measures the overall proportion of correct predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Here, TP (True Positives) refers to correctly predicted offensive tweets, FP (False Positives) are non-offensive tweets incorrectly predicted as offensive, FN (False Negatives) are offensive tweets missed by the model, and TN (True Negatives) are correctly predicted non-offensive tweets.

IV. Results

Each LLM was evaluated under three prompting configurations: zero-shot, one-shot, and few-shot. In the zero-shot setting, models were asked to classify tweets without any examples. The one-shot and few-shot settings included one and three labeled examples, respectively, embedded within the prompt. All models received the same classification prompt format to ensure consistency, and their predictions were compared against human-annotated labels.

Table III presents the performance of all evaluated models: AraBERT, LLaMA3, Gemma2, GPT-4o, and ALLaM on the Arabic hate speech classification task. We report Accuracy, Precision, Recall, and F1-score. All LLMs were able to outperform the baseline (AraBERT) across all metrics with F1-scores ranging from 0.75 to 0.86, compared to AraBERT's 0.71. GPT-4o consistently achieved the highest performance across all metrics and shot settings, with its F1-score improving from 0.83 (zero-shot) to 0.86 (three-shot). LLaMA3 and Gemma2 also demonstrated strong performance, showing consistent gains with the additional in-context examples. As the only Arabic-specific model in our evaluation, ALLaM started with a lower F1-score (0.75) in zero-shot, but improved significantly to 0.82 in the three-shot setup.

To better understand the level of agreement between the top-performing models, GPT-4o, LLaMA3, Gemma2, and ALLaM, we computed Fleiss' Kappa, a statistical measure of inter-rater reliability designed for evaluating consistency across multiple raters or classifiers [36]. A resulting score of 0.726 indicates substantial agreement, with values above 0.60 generally considered strong [37], suggesting that despite architectural and training differences, these LLMs produce largely consistent classification outcomes. As a result, the potential benefits of ensembling are reduced, since high agreement implies limited diversity in model outputs. Indeed, a majority-vote ensemble of the four models yielded performance nearly identical to GPT-4o alone (F1 = 0.86), reinforcing the conclusion that GPT-4o already captures most of the predictive power available across models. This finding aligns with recent studies, which report that ensemble methods offer the greatest benefit when base models are diverse in their predictions, and that high inter-model agreement can limit the effectiveness of majority-vote ensembles in NLP tasks [38], [39].

To assess the impact of class distribution on LLMs performance on the task at hand, we also evaluated all models on an imbalanced sample 1160:439, which reflects the original class distribution found in the dataset. This comparison provides insight into model robustness and real-world applicability. Table IV summarizes these results, with all metrics reported as macro-averages to ensure fair comparison across classes.

As expected, overall accuracy increased for most models compared to the balanced setting, since the majority class dominates the prediction task. However, macro F1-scores and recall values did not always improve, highlighting the challenges of maintaining balanced performance across all classes when the data is skewed. GPT-4o maintained the strongest results across all settings, achieving a macro F1-score of 0.84 in both the zero-shot and three-shot configurations. LLaMA3 and ALLaM also performed well in the three-shot setting, with F1-scores of 0.82 and 0.79, respectively. Gemma2's performance varied

TABLE II. PROMPT DESIGN EXAMPLES FOR 0-SHOT AND FEW-SHOT SETTINGS

0-shot Prompt	**Few-shot Prompt (e.g., 1-shot)**

Classify the input Arabic tweet as 'offensive' or 'not'.
The text will be delimited by triple backticks.
Answer only with 'offensive' or 'not'. Do not explain your answer!
``` tweet content ```
	Classify the input Arabic tweet as 'offensive' or 'not'.
	The text will be delimited by triple backticks.
	Answer only with 'offensive' or 'not'.
	Examples:
	``` Example tweet 1 ``` → offensive
	``` Example tweet 2 ``` → not
	Now classify:
	``` tweet content ```

TABLE III. PERFORMANCE OF LLMS ON THE (BALANCED) DATASET

Model		**Accuracy**	**Precession**	**Recall**	**F1 score**
AraBERT		0.71	0.72	0.71	0.71
LLaMA3	0-shot	0.79	0.80	0.79	0.79
	1-shot	0.82	0.83	0.82	0.82
	3-shot	0.83	0.84	0.83	0.83
Gemma2	0-shot	0.79	0.82	0.79	0.79
	1-shot	0.80	0.82	0.80	0.80
	3-shot	0.82	0.83	0.82	0.82
GPT-4o	0-shot	0.83	0.84	0.83	0.83
	1-shot	0.85	0.87	0.85	0.85
	3-shot	**0.86**	**0.87**	**0.86**	**0.86**
ALLaM	0-shot	0.76	0.79	0.76	0.75
	1-shot	0.78	0.82	0.78	0.78
	3-shot	0.82	0.82	0.81	0.81

more across shots, but still benefited from the addition of more in-context examples. It reached an F1-score of 0.79 in the three-shot setting despite lower scores at zero- and one-shot settings. AraBERT, serving as the baseline, achieved a macro F1-score of 0.70, which was outperformed by all LLMs.

TABLE IV. PERFORMANCE OF LLMS ON THE (IMBALANCED) DATASET. ALL METRICS ARE IN MACRO-AVERAGE

Model		**Accuracy**	**Precision**	**Recall**	**F1 score**
AraBERT		0.75	0.69	0.71	0.70
LLaMA3	0-shot	0.85	0.81	0.80	0.80
	1-shot	0.81	0.77	0.83	0.79
	3-shot	0.85	0.81	0.83	0.82
Gemma2	0-shot	0.74	0.74	0.80	0.72
	1-shot	0.71	0.73	0.79	0.70
	3-shot	0.81	0.77	0.83	0.79
GPT-4o	0-shot	**0.87**	**0.83**	0.85	**0.84**
	1-shot	0.84	0.80	**0.87**	0.82
	3-shot	0.86	0.83	0.85	**0.84**
ALLaM	0-shot	0.84	0.81	0.77	0.78
	1-shot	0.78	0.76	0.82	0.76
	3-shot	0.82	0.78	0.82	0.79

These findings confirm that LLMs are generally robust even under imbalanced conditions, particularly when few-shot prompts are provided. However, they also reinforce the importance of reporting macro-averaged metrics in hate speech classification tasks, as high accuracy alone may mask poor performance on the minority class (e.g. offensive content). While the overall performance of LLMs was strong, a closer examination of misclassifications provides further insight into model behavior, as discussed in the following section.

V. DISCUSSION

Our findings shows that LLMs are highly effective for Arabic hate speech detection, with GPT-4o achieving the highest F1-score (0.86) in the three-shot setting. This strong performance reflects GPT-4o's multilingual generalization abilities, as documented in prior work [27]. Additionally, open-source models like LLaMA3 and Gemma2 performed competitively, especially when guided with well-structured prompts, demonstrating that they are suitable for Arabic NLP tasks [40]. Moreover, the improvement in ALLaM's F1-score from 0.75 (zero-shot) to 0.82 (three-shot) aligns with patterns observed in other Arabic-centric models, such as JAIS [12], where the inclusion of contextual examples substantially narrows the gap between pretraining objectives and downstream task performance.

These findings suggest that prompt-based learning strategies are particularly effective for hate speech detection in Arabic, a domain where linguistic nuance and cultural context play a significant role in defining classification boundaries [15]. The observed effectiveness of in-context examples supports theoretical frameworks proposing that LLMs are capable of capturing implicit sociolinguistic cues, such as sarcasm and dialectal variation, when provided with representative examples [41], [42]. This is especially critical in Arabic, where hate speech often appears through indirect references and dialect-specific expressions that pose challenges to static lexicon-based approaches [15]. However, a deeper analysis of model errors is essential to understand the nature and sources of misclassifications, and to guide future improvements, especially in the context of Saudi Arabia.

To gain deeper insights into model behavior, we conducted a comprehensive error analysis combining confusion matrices and manual inspection of misclassified instances. Fig. 2 displays the confusion matrices for all models under the three-shot setting using the balanced dataset, while Fig. 3 illustrates performance on the imbalanced dataset. These visualizations highlight distinct error patterns shaped by class distribution and reveal how each model handles nuanced classification scenarios.

The first category of error involves class-specific misclassification patterns. The confusion matrices reveal that the distribution of errors differs significantly between the balanced and imbalanced datasets. In the balanced setting, models tend to generate more false positives than false negatives. For instance, GPT-4o misclassified 17.5% of non-offensive tweets as offensive (140 out of 800), whereas its false negative rate was 9.9% (79 out of 800 offensive tweets misclassified as non-offensive). However, this trend is reversed in the imbalanced

setting: GPT-4o's false negative rate increases to 17.1% (75 out of 439), while false positives drop to 12.1% (141 out of 1161). This shift illustrates the model's tendency to favor the majority class in skewed distributions, a common issue in imbalanced classification tasks [41], [17].

The second category of error is Model-Specific Vulnerabilities. Gemma2 showed the most notable sensitivity to data distribution, with a consistently high false positive rate, 27.3% in the balanced setting (218 out of 800) and 21.1% in the unbalanced setting (245 out of 1161). This pattern suggests a persistent tendency to misclassify non-offensive content as offensive, possibly due to limitations in handling subtle contextual cues. In contrast, ALLaM demonstrated more stable behavior across distributions, with false positive rates of 23.8% in the balanced setting (190 out of 800) and 18.1% in the unbalanced setting (210 out of 1161). This relatively moderate error fluctuations indicates stronger fine-tuning for Arabic hate speech detection despite having slightly lower overall accuracy than GPT-4o or LLaMA3.

The Third category of error stems from linguistic and contextual challenges. To further investigate these challenges, we conducted a manual error analysis on a stratified sample of 60 misclassified tweets, with 15 instances drawn from each of the three-shot settings of the top-performing models, i.e. GPT-4o, LLaMA3, Gemma2, and ALLaM. Each misclassified tweet was annotated with a single dominant error category based on linguistic, semantic, or contextual cues that likely contributed to the misclassification. Fig. 4 illustrates the distribution of error categories among the misclassified sample.

The most common error category involved tweets expressing implicit hate and indirect language, which accounted for 30% of the annotated sample, including 7% explicitly marked as sarcasm. These tweets often relied on cultural references, regional superiority, or subtle phrasing rather than directly offensive language. For example, the statement أنا من أهل [..] لم أسمع بهذه العادة من قبل! من أين تأتون بها؟ ("I'm from the [..] region, and I've never heard of this custom before. Where are you bringing it from?") implicitly conveys cultural exclusion and superiority without directly attacking a group. By framing the custom as unfamiliar or foreign, the speaker questions its authenticity and implicitly discriminates against those who practice it. Such instances of indirect hate pose a challenge for large language models, which often rely on explicit lexical indicators. Detecting these subtle nuances requires deeper context awareness and cultural understanding, areas where current models still face limitations.

Another major source of error is dialect ambiguity observed in 20% of the sample. Tweets written in regional dialects such as Najdi or Hijazi, or those incorporating local slang, were frequently misclassified. This is likely due to the under-representation of these dialects in most LLM pretraining corpora. Dialectal differences in vocabulary, syntax, and idiomatic expressions can significantly alter the perceived offensiveness of a statement, particularly when models lack targeted exposure to these variations. Additionally, some expressions observed in the dataset reflected linguistic influence from neighboring regions, such as Gulf and Egyptian dialects, which are naturally present in the Saudi linguistic landscape. This cross-regional blending adds further complexity to model interpretation, especially when the cues for offensiveness are subtle or culturally specific.

Misclassifications also occurred in 17% of tweets that mentioned named individuals or groups without providing sufficient contextual cues. This type of named entity confusion arises when a model lacks the external or background knowledge needed to determine whether a reference is offensive or neutral. In many cases, tweets rely on insider knowledge, local discourse, or references to public figures, making it difficult for models, particularly in zero- or few-shot settings, to correctly interpret intent. For instance, the tweet "بكاء الحقوقيات طرب" ("The crying of feminists is like music") may appear neutral or poetic. However, depending on the social context, it may carry sarcasm or targeted implications. Another tweet, "رواه البخاري ومسلم", references a well-known hadith phrase, but when used outside religious discourse, it can be employed mockingly or to dismiss opposing views. Therefore without the relevant cultural grounding, models may either over- or under-classify such tweets, leading to errors. This highlights the limitation of LLMs in performing context-aware reasoning and culturally informed classification in low-context tasks.

Another challenge observed in 10% of the sample was topic bias. This occurs when the model misclassifies tweets as offensive or non-offensive based primarily on the subject matter or specific keywords, rather than the actual tone or context. Tweets discussing sensitive or controversial topics, such as politics, religion, or social issues, may be flagged as offensive even when they are neutral or factual. For instance, terms like اليهود, and إدمان ("jewish, and addiction") can trigger false positives due to their frequent association with offensive content during training. A relevant example is: "مع ايقاف اللعبة وبشده صرنا مدمنين عليها بشكل" ("I strongly support banning this game … we've become terribly addicted to it."), which is factually neutral but was incorrectly flagged as offensive. However, tweets containing harmful language on less controversial topics may escape detection if the model has implicitly learned to associate those subjects with with non-offensive speech. This bias is due to the model overfitting to patterns in the training data, where certain topics are far less represented compared to offensive speech.

The final category of errors relates to annotation ambiguity, accounting for 10% of the sample. In these cases, the ground truth labels themselves appeared questionable or inconsistent. This reflects the inherent subjectivity involved in labeling hate or offensive speech, especially in borderline cases or when annotators apply differing cultural or personal interpretations. Such ambiguity introduces label noise, which not only affects model training but also complicates evaluation by obscuring what constitutes a correct prediction. For example, the tweet "يا[–] جيزان عسير نجران أراضي [–]" ("[Group] [Region names] belong to [Country]") was annotated as not offensive. However, it arguably conveys an implicit denial of national identity, raising concerns about annotation consistency. This highlights how politically sensitive claims, when expressed implicitly, may be overlooked by annotators, reinforcing the importance of clear annotation criteria and sociopolitical awareness in training data curation.

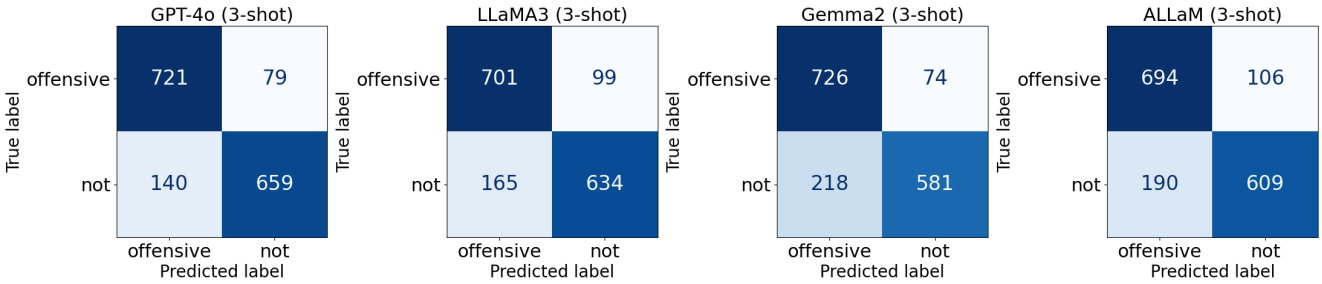


Fig. 2. Confusion matrices of LLMs under three-shot setting on the balanced dataset.

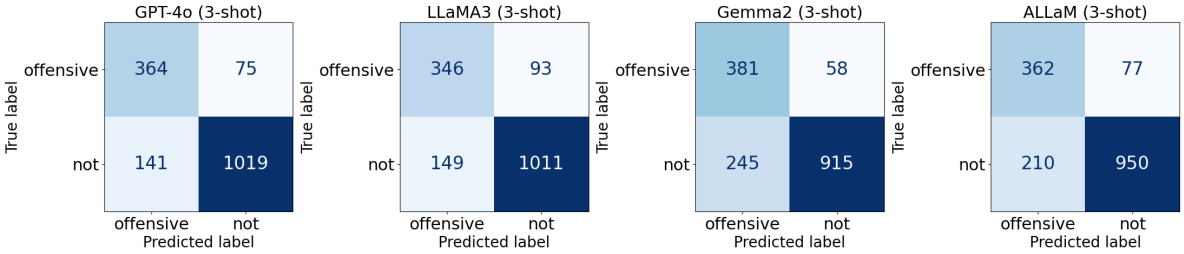


Fig. 3. Confusion matrices of LLMs under three-shot setting on the unbalanced dataset.

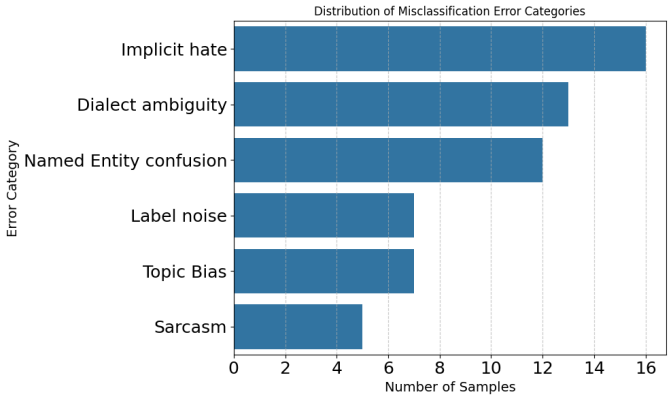


Fig. 4. The distribution of misclassifications error categories.

Our analysis reveals several persistent challenges that extend beyond technical limitations. These challenges are deeply rooted in the sociolinguistic complexity of Arabic, as well as in the subjective nature of annotating hate speech. Addressing such issues requires both linguistic sensitivity and culturally aware annotation strategies.

VI. Conclusion

This study examined the capabilities of LLMs in classifying Arabic hate speech, with a particular focus on prompt-based learning and sociolinguistic differences within Saudi dialects. The evaluation was performed under different prompting settings, including zero-shot, one-shot, and three-shot. While GPT-4o outperformed all models, open-source models such as LLaMA3, Gemma2, and ALLaM also demonstrated competitive performance. Moreover, all models benefited from the support of relevant contextual examples. A key finding is that well-structured prompts significantly enhance model performance, further supporting the potential of prompt engineering in low-resource language settings.

Our in-depth error analysis revealed that misclassifications stem from a mix of technical and sociolinguistic challenges. Common sources of error included implicit hate, dialectal variation, named entity confusion, topic bias, and annotation ambiguity. These findings highlight the need for culturally informed evaluation, richer dialectal representation, and improved annotation guidelines in Arabic NLP tasks. While the evaluation was conducted on a representative subset of the dataset, this approach aligns with established practices in LLM assessment and offers reliable insights into model behavior across larger corpora.

Beyond benchmarking, improved hate speech detection in Arabic dialects can support safer online spaces, strengthen content moderation systems, and inform more nuanced policy-making in multilingual contexts. Addressing annotation ambiguity also calls for culturally grounded guidelines and tools that better support annotators. Future research should explore hybrid modeling approaches, culturally enriched training data, dialect-specific fine-tuning, and more representative few-shot examples to further enhance prompt-based performance.

References

[1] A. Alhazmi, R. Mahmud, N. Idris, M. E. M. Abo, and C. Eke, "A systematic literature review of hate speech identification on arabic twitter data: research challenges and future directions," *PeerJ Computer Science*, vol. 10, p. e1966, 2024.

[2] A. Omar, T. M. Mahmoud, and T. Abd-El-Hafeez, "Comparative performance of machine learning and deep learning algorithms for arabic hate speech detection in osns," in *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*. Springer, 2020, pp. 247–257.

[3] M. Hietanen and J. Eddebo, "Towards a definition of hate speech—with a focus on online contexts," *Journal of Communication Inquiry*, vol. 47, no. 4, pp. 440–458, 2023.

[4] UNESCO, "World arabic language day," n.d., accessed: 2025-06-10. [Online]. Available: https://www.unesco.org/ar/world-arabic-language-day

[5] A. Alakrot, L. Murray, and N. S. Nikolov, "Dataset construction for the detection of anti-social behaviour in online communication in arabic," in *Procedia Computer Science*, vol. 142. Elsevier, 2018, pp. 174–181.

[6] A. Charfi, M. Besghaier, R. Akasheh, A. Atalla, and W. Zaghouani, "Hate speech detection with adhar: a multi-dialectal hate speech corpus in arabic," vol. 7. Frontiers Media SA, 2024, p. 1391472.

[7] T. Alhindi, P. Ghaffari, and W. Magdy, "Fine-tuned hate speech detection in arabic social media," *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 20, no. 2, pp. 1–19, 2021.

[8] OpenAI, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[9] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[10] M. Zouidine and M. Khalil, "Large language models for arabic sentiment analysis and machine translation," *Engineering, Technology & Applied Science Research*, vol. 15, no. 2, pp. 20 737–20 742, 2025.

[11] A. Atef, F. Seddik, and A. Elbedewy, "Ags: Arabic gpt summarization corpus," in *2023 International Conference on Electrical, Communication and Computer Engineering (ICECCE)*. IEEE, 2023, pp. 1–8.

[12] A. A. Al Wazrah, A. Altamimi, H. Aljasim, W. Alshammari, R. Al-Matham, O. Elnashar, M. Amin, and A. AlOsaimy, "Evaluation of large language models on arabic punctuation prediction," in *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, 2025, pp. 144–154.

[13] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A bert-based transfer learning approach for hate speech detection in online social media," in *International conference on complex networks and their applications*. Springer, 2019, pp. 928–940.

[14] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," *arXiv preprint arXiv:2003.00104*, 2020.

[15] H. Mubarak, S. Hassan, and S. A. Chowdhury, "Emojis as anchors to detect arabic offensive language and hate speech," *Natural Language Engineering*, vol. 29, no. 6, pp. 1436–1457, 2023.

[16] M. Tonneau, D. Liu, S. Fraiberger, R. Schroeder, S. Hale, and P. Röttger, "From languages to geographies: Towards evaluating cultural bias in hate speech datasets," in *Proceedings of the 8th Workshop on Online Abuse and Harms (WOAH 2024)*. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 283–311. [Online]. Available: https://aclanthology.org/2024.woah-1.23

[17] A. Ahmad, M. Azzeh, E. Alnagi, Q. Abu Al-Haija, D. Halabi, A. Aref, and Y. AbuHour, "Hate speech detection in the arabic language: corpus design, construction, and evaluation," *Frontiers in Artificial Intelligence*, vol. 7, p. 1345445, 2024.

[18] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the saudi twittersphere," *Applied Sciences*, vol. 10, no. 23, p. 8614, 2020.

[19] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024.

[20] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "Arbert & marbert: Deep bidirectional transformers for arabic," *arXiv preprint arXiv:2101.01785*, 2020.

[21] N. Wies, Y. Levine, and A. Shashua, "The learnability of in-context learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 36 637–36 651, 2023.

[22] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[23] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[24] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé *et al.*, "Gemma 2: Improving open language models at a practical size," *arXiv preprint arXiv:2408.00118*, 2024.

[25] M. S. Bari, Y. Alnumay, N. A. Alzahrani, N. M. Alotaibi, H. A. Alyahya, S. AlRashed, F. A. Mirza, S. Z. Alsubaie, H. A. Alahmed, G. Alabduljabbar *et al.*, "Allam: Large language models for arabic and english," *arXiv preprint arXiv:2407.15390*, 2024.

[26] P. Dadure, A. Dixit, K. Tewatia, N. Paliwal, and A. Malla, "Sentiment analysis of arabic tweets using large language models," in *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, 2025, pp. 88–94.

[27] S. Khaled, E. H. Mohamed, and W. Medhat, "Evaluating large language models for arabic sentiment analysis: A comparative study using retrieval-augmented generation," *Procedia Computer Science*, vol. 244, pp. 363–370, 2024.

[28] S. Alghamdi, Y. Benkhedda, B. Alharbi, and R. T. Batista-Navarro, "Aratar: A corpus to support the fine-grained detection of hate speech targets in the arabic language," in *Proceedings of the 6th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT) with Shared Tasks on Arabic LLMs Hallucination and Dialect to MSA Machine Translation@ LREC-COLING 2024*, 2024, pp. 1–12.

[29] T. Alqurashi, "Applying a character-level model to a short arabic dialect sentence: a saudi dialect as a case study," *Applied Sciences*, vol. 12, no. 23, p. 12435, 2022.

[30] K. Madukwe, X. Gao, and B. Xue, "In data we trust: A critical analysis of hate speech detection datasets," in *Proceedings of the fourth workshop on online abuse and harms*, 2020, pp. 150–161.

[31] D. Ramachandran and R. Parvathi, "Analysis of twitter specific preprocessing technique for tweets," *Procedia Computer Science*, vol. 165, pp. 245–251, 2019.

[32] S. Ruder, A. Søgaard, and I. Vulić, "Unsupervised cross-lingual representation learning," in *Proceedings of the 57th annual meeting of the association for computational linguistics: Tutorial abstracts*, 2019, pp. 31–38.

[33] G. Inc. (2024) Groq api documentation. Accessed May 12, 2025. [Online]. Available: https://docs.groq.com

[34] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

[35] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[36] K. L. Gwet, "Large-sample variance of fleiss generalized kappa," *Educational and Psychological Measurement*, vol. 81, no. 4, pp. 781–790, 2021.

[37] F. Moons and E. Vandervieren, "Measuring agreement among several raters classifying subjects into one-or-more (hierarchical) nominal categories. a generalisation of fleiss' kappa," *arXiv preprint arXiv:2303.12502*, 2023.

[38] H. Mubarak, K. Darwish, W. Magdy, T. Elsayed, and H. Al-Khalifa, "Overview of osact4 arabic offensive language detection shared task," in *Proceedings of the 4th Workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection*, 2020, pp. 48–52.

[39] K. E. Daouadi, Y. Boualleg, and K. E. Haouaouchi, "Ensemble of pre-trained language models and data augmentation for hate speech detection from arabic tweets," *arXiv preprint arXiv:2407.02448*, 2024.

[40] M. T. I. Khondaker, N. Naeem, F. Khan, A. Elmadany, and M. Abdul-Mageed, "Benchmarking llama-3 on arabic language generation tasks," in *Proceedings of The Second Arabic Natural Language Processing Conference*, 2024, pp. 283–297.

[41] K. E. Daouadi, Y. Boualleg, and O. Guehairia, "Comparing pre-trained language model for arabic hate speech detection," *Computación y Sistemas*, vol. 28, no. 2, pp. 681–693, 2024.

[42] J. M. Pérez, P. Miguel, and V. Cotik, "Exploring large language models for hate speech detection in rioplatense spanish," *arXiv preprint arXiv:2410.12174*, 2024.

Task-Oriented Evaluation of Assamese Tokenizers Using Sentiment Classification

Basab Nath[1]*, Sagar Tamang[2], Osman Elwasila[3], Yonis Gulzar[4]*
School of Computer Science and Engineering, Bennett University, Greater Noida, India[1]
Centre for Educational Technology, Indian Institute of Technology Patna, Bihar 835217, India[2]
Department of Management Information Systems-College of Business Administration,
King Faisal University, Al-Ahsa 31982, Saudi Arabia[3,4]

***Abstract*—Tokenization is a foundational step in the NLP pipeline, and its design strongly influences the performance of transformer-based models, particularly for morphologically rich and low-resource languages such as Assamese. While most tokenizers are traditionally assessed using intrinsic metrics, their practical impact on downstream tasks has remained underexplored. This study systematically evaluates nine subword tokenizer configurations—spanning Byte-Pair Encoding (BPE), WordPiece, and Unigram algorithms with vocabulary sizes of 8K, 16K, and 32K—on sentiment classification in Assamese. Each tokenizer was integrated into a BERT-base-multilingual-cased model by replacing the default tokenizer and reinitializing the embedding layer. On a manually curated dataset, naïve fine-tuning proved unstable under class imbalance, but a class-weighted loss restored effective training and exposed clear performance differences across tokenizers. WordPiece consistently outperformed BPE and Unigram, with the wordpiece_16k configuration achieving a weighted F1-score of 0.4897 across 10 random seeds. This score was statistically comparable to mBERT (0.4919) and competitive with larger multilingual baselines such as XLM-R (0.4978), despite relying on a far smaller, Assamese-specific vocabulary. These findings underscore that tokenizer choice is not a neutral preprocessing step but a critical design decision, highlighting the importance of downstream evaluation when developing practical NLP pipelines for low-resource languages.**

***Keywords*—Assamese NLP; tokenization; subword tokenization; sentiment analysis; low-resource languages; BERT; class imbalance**

I. Introduction

Tokenization is a foundational step in the Natural Language Processing (NLP) pipeline, particularly for transformer-based architectures that operate at the subword level. By segmenting raw text into manageable units, tokenizers influence vocabulary coverage, sequence length, and ultimately task performance [1], [2]. Prior research has also shown that structural representation choices, such as graph-based modeling of dependencies, can significantly affect semantic interpretation and downstream performance [3]. While much work has emphasized the development of efficient tokenization algorithms, relatively little attention has been paid to their impact on downstream tasks in low-resource scenarios. This gap is especially pronounced for Indic languages such as Assamese, where linguistic complexity intersects with data scarcity.

*Corresponding Authors, emails: biyau@iuiu.ac.ug, ygulzar@kfu.edu.sa

Assamese, spoken by more than 15 million people, is a morphologically rich and underrepresented language in NLP research. Its script shares similarities with Bengali but includes unique graphemes, creating additional tokenization challenges. The language also exhibits complex inflectional and derivational morphology, with affixes encoding case, tense, aspect, and honorific markers. Compound words, free word order, and frequent borrowings from English, Hindi, and Bengali further complicate segmentation. Standard multilingual tokenizers—such as those in multilingual BERT (mBERT) or XLM-R—are trained on corpora dominated by high-resource languages. Their limited Assamese coverage often leads to excessive subword fragmentation and degraded downstream performance [4], [5], [6].

Rust et al. [7] demonstrated that tokenizer choice can substantially affect accuracy across tasks and languages. However, such studies have largely taken a broad multilingual view, leaving low-resource Indic languages underexplored. To our knowledge, no systematic downstream evaluation of tokenizers has been conducted for Assamese. Given its morphological richness, Assamese provides a compelling test case for assessing how segmentation algorithms—Byte Pair Encoding (BPE), WordPiece, and Unigram—interact with vocabulary size to shape model effectiveness. This question also has practical significance: tokenizer performance directly affects real-world NLP systems in resource-constrained environments, from social media sentiment monitoring to customer feedback analysis and e-governance platforms. Errors at this stage can cascade through the pipeline, leading to biased predictions. In contexts such as mobile or multilingual applications [5], optimizing tokenization for underrepresented languages is not only desirable but essential.

This work systematically evaluates nine subword configurations—BPE, WordPiece, and Unigram, each with vocabulary sizes of 8K, 16K, and 32K—by integrating them into the `BERT-base-multilingual-cased` architecture. Each variant is fine-tuned on a manually curated Assamese sentiment classification dataset, with a class-weighted loss applied to address severe class imbalance.

This study is guided by the following research questions:

- How does the choice of tokenization algorithm (BPE, WordPiece, Unigram) affect downstream sentiment classification in Assamese?
- What role does vocabulary size play in balancing coverage and performance?

- Do custom Assamese tokenizers offer tangible improvements over off-the-shelf multilingual tokenizers such as the default mBERT tokenizer or IndicBERT?
- What qualitative differences in segmentation behavior help explain the performance disparities?

The contributions of this paper are fourfold. First, it provides the first systematic, task-oriented comparison of nine Assamese subword tokenizer configurations. Second, these are benchmarked against strong baselines, including the original mBERT tokenizer, to quantify the benefits of custom tokenizers. Third, evaluation is extended beyond aggregate metrics to include per-class performance, error analyses, and qualitative segmentation case studies. Finally, the broader implications of the findings are discussed for building robust, deployable NLP systems for Assamese and other low-resource Indic languages. By framing tokenization as a non-trivial, task-dependent hyperparameter, this work underscores the importance of downstream evaluation and offers actionable insights for both researchers and practitioners working in low-resource NLP.

II. Related Work

Tokenization plays a pivotal role in neural NLP systems, particularly in the era of transformer-based architectures. Early work by Sennrich et al. [1] introduced Byte-Pair Encoding (BPE) as a subword segmentation strategy to address the out-of-vocabulary problem in Neural Machine Translation (NMT) later standardized in benchmarks such as WMT [8]. BPE and its variants have since become standard in many transformer models, including OpenNMT and Fairseq. Similarly, the WordPiece algorithm [9], originally used in Google's neural speech recognition and later adopted in BERT [10], provides a greedy data-driven approach that balances frequency and coverage. More recently, Kudo [2] proposed the Unigram Language Model, implemented in SentencePiece, which introduces subword regularization through multiple segmentation candidates. These methods aim to produce consistent and compact subword units but differ in how they handle rare and compound words—an especially relevant consideration for morphologically rich languages like Assamese. Despite the prevalence of intrinsic tokenizer evaluation (e.g. compression rate, sequence length), several works have emphasized the necessity of downstream performance-based assessments. Rust et al. [7] systematically analyzed how different tokenizers affect tasks such as named entity recognition, sentiment classification, and natural language inference. Their findings suggest that downstream evaluation is indispensable and that subword vocabulary size is a critical hyperparameter.

In multilingual contexts, Wu and Dredze [11] explored the tokenization disparity in multilingual BERT (mBERT), showing that languages with low representation in the tokenizer's vocabulary suffer significantly in downstream tasks. Similarly, Wang et al. [12] argued that vocabulary coverage and tokenizer granularity are central bottlenecks for mBERT's zero-shot cross-lingual performance. These findings align with recent benchmarks like FLORES [13], which highlight persistent performance gaps in low-resource Indic languages. Several studies have targeted the Indic NLP domain. Bhattacharjee et al. [14] introduced the IndicCorp and Samanantar corpora, which have become foundational for training and evaluating multilingual models on Indian languages. Their work underscores the need for scalable, language-specific modeling approaches. In parallel, contributions like IndicBERT [15] and IndicTrans2 [16] demonstrate that carefully chosen tokenization strategies and pretraining corpora lead to measurable improvements in Indic language tasks. Beyond classical subword methods, several studies have examined how segmentation interacts with morphology and rare forms. Boström and Durrett [17] showed that byte-level segmentation can reduce the brittleness of BPE on morphologically rich words, while Provilkov et al. [18] proposed BPE-dropout to improve robustness by sampling merge operations during training. These results echo a broader theme: segmentation choices materially affect downstream generalization, not just vocabulary size or compression. Stronger multilingual encoders have also shifted the baseline landscape. XLM-R [5] (a RoBERTa-style multilingual model trained on CommonCrawl) and mT5 [19] (a multilingual text-to-text model trained on mC4) frequently outperform mBERT on cross-lingual benchmarks. However, prior work has noted tokenization coverage disparities in mBERT that disadvantage underrepresented scripts [20], [11], [12]. Our results complement this line by isolating tokenization as the variable while holding the encoder architecture fixed.

Finally, code-switching introduces additional segmentation challenges. Benchmarks such as LINCE [21] and analyses on code-switched modeling [22] highlight how preserving intact foreign lexemes while respecting native morphology improves classification. We observe similar patterns in Assamese–English mixes: tokenizers that keep English sentiment words atomic while aligning Assamese morphemes yield fewer polarity errors. Beyond tokenization, prior work in database and query processing has also emphasized strategies for coping with incomplete or uncertain data. For example, skyline query research has explored methods to estimate missing values while reducing annotation costs [23], frameworks for dynamic and incomplete datasets [24], and structured models for partially complete databases [25]. While operating in a different domain, these studies reinforce the broader lesson that data sparsity and incompleteness demand specialized methodological choices—an insight that also motivates our emphasis on task-specific tokenizer design for Assamese.

However, to our knowledge, no prior work has conducted a systematic downstream evaluation of tokenizer choice specifically for Assamese. Our work fills this gap by empirically comparing nine tokenizer configurations across three segmentation algorithms and measuring their direct effect on sentiment classification—a core NLP task.

III. Methodology

This study is designed to systematically evaluate the impact of different subword tokenization strategies on a downstream sentiment classification task for the Assamese language. A two-stage experimental process was adopted. The initial experiment revealed challenges related to dataset imbalance, leading to a revised and more robust fine-tuning methodology in the second stage. Additionally, to ensure comprehensive evaluation, the custom tokenizers were benchmarked against strong baselines, namely the default mBERT tokenizer and

the tokenizer used in IndicBERT, to verify whether Assamese-specific vocabularies provide tangible benefits.

A. Dataset

Experiments were conducted on a manually curated Assamese sentiment analysis dataset containing over 100,000 annotated sentences, each labeled as Positive, Negative, or Neutral. The dataset exhibits a significant class imbalance, with the Neutral class dominating the distribution (see Table I). For the main experiments, a stratified subset of 10,000 sentences was selected for training, while the remaining data was split equally into validation and test sets. Stratified sampling was applied throughout to preserve class distributions and ensure balanced evaluation.

TABLE I. CLASS DISTRIBUTION IN THE ASSAMESE SENTIMENT DATASET

Sentiment Class	Percentage	Count
Neutral	61.80%	74,730
Positive	27.16%	32,844
Negative	11.03%	13,347

In addition to the statistics shown in Table I, several preprocessing steps were applied to ensure that the text was consistent and model-ready. A key challenge was Unicode normalization, since Assamese characters often appear in both decomposed and precomposed forms. For example, the sequence "ক + ় " was normalized into the single grapheme কয়. Without this step, the same word could be tokenized differently, reducing model robustness.

Noisy characters such as emojis and rare punctuation were removed. For instance, the raw tweet এই ছবিখন ভাল নহয় was simplified to এই ছবিখন ভাল নহয়. Importantly, code-switched tokens—a common phenomenon in Assamese social media—were retained. A typical case is movie খুব বৰিনগ অসিল, where the English word "boring" was preserved, since removing it would distort the semantic content.

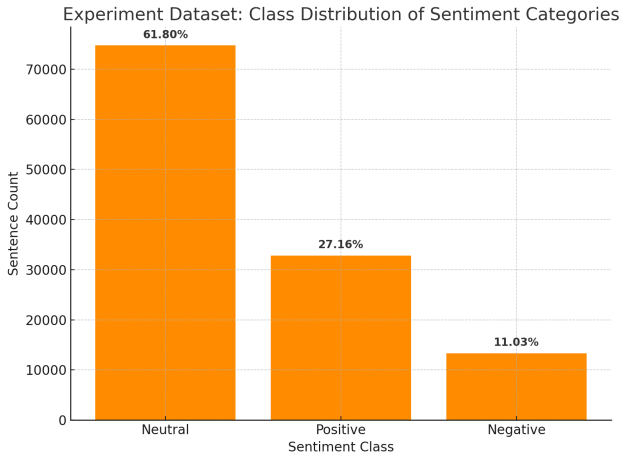


Fig. 1. Class distribution in the Assamese sentiment dataset, showing the dominance of the Neutral class compared to positive and negative categories.

Fig. 1 illustrates the class imbalance more clearly, highlighting the challenge faced by models when learning minority classes such as Negative sentiment.

B. Tokenizer Training

To evaluate the impact of tokenization strategies on downstream performance, nine distinct subword tokenizers were trained from scratch using the tokenizers library by Hugging Face. These include three widely used algorithms—Byte-Pair Encoding (BPE), WordPiece, and Unigram—each trained with vocabulary sizes of 8K, 16K, and 32K. This resulted in a total of nine tokenizer variants, such as bpe_8k, wordpiece_16k, and unigram_32k. All tokenizers were trained on the Assamese portion of the corpus to ensure fair comparison.

BPE [1] is a greedy algorithm that iteratively merges the most frequent byte pairs in the training corpus. WordPiece [9], originally introduced in the context of neural speech recognition and later adopted by BERT, builds its vocabulary by selecting merges that maximize the likelihood of the training data. Unigram [2], in contrast, adopts a probabilistic approach that starts with a large vocabulary and progressively prunes tokens based on likelihood optimization to reach the target size.

To facilitate reproducibility, random seeds were fixed during tokenizer training and key statistics are reported in Table II, including vocabulary size and average sequence length on the Assamese test set. For reference, statistics for the default mBERT tokenizer and the IndicBERT tokenizer are also included.

C. Sentiment Classification Model

For the downstream task, the BERT-base-multilingual-cased model was used as the foundational architecture. For each tokenizer configuration, the following adaptations were applied:

- A fresh instance of the pre-trained BERT-base-multilingual-cased model was loaded.
- The model's default WordPiece tokenizer was replaced with one of the custom-trained Assamese tokenizers (or left unchanged for baseline runs).
- The token embedding layer was resized using resize_token_embeddings() to match the vocabulary size of the new tokenizer. Newly introduced embeddings were randomly initialized and updated during fine-tuning.

Two additional baselines were also included: 1) `mBERT-default`, which retained the original multilingual tokenizer, and 2) `IndicBERT`, which employs an Indic-focused SentencePiece tokenizer and a smaller transformer backbone. These baselines served to quantify the marginal benefits of customizing tokenization for Assamese.

D. Fine-Tuning Procedure

Model fine-tuning was conducted in two distinct experimental phases.

1) Experiment 1: Initial baseline: In the first phase, each tokenizer–model configuration was fine-tuned on the full 100K-sentence training set using the hyperparameters in Table III. This established a baseline for subsequent refinements.

TABLE II. TOKENIZER CONFIGURATIONS AND STATISTICS. AVERAGE LENGTH IS THE MEAN NUMBER OF TOKENS PER SENTENCE ON THE ASSAMESE TEST SET

Tokenizer Variant	Algorithm	Vocab Size	Training Strategy	Avg. Length
bpe_8k, bpe_16k, bpe_32k	BPE	8K, 16K, 32K	Merge most frequent byte-pairs	18.7 / 16.3 / 15.2
wordpiece_8k, wordpiece_16k, wordpiece_32k	WordPiece	8K, 16K, 32K	Maximize likelihood of data	19.1 / 15.6 / 15.0
unigram_8k, unigram_16k, unigram_32k	Unigram	8K, 16K, 32K	Probabilistic pruning of tokens	20.3 / 18.4 / 17.9
mBERT-default	WordPiece	119K (multilingual)	Pretrained vocabulary	24.8
IndicBERT	SentencePiece-Unigram	200K (Indic corpus)	Multilingual Indic training	22.5

TABLE III. HYPERPARAMETERS USED FOR THE INITIAL BASELINE EXPERIMENT

Hyperparameter	Value
Epochs	3
Batch Size	16
Learning Rate	2×10^{-5}
Optimizer	AdamW ($\epsilon = 1 \times 10^{-8}$)
Max Sequence Length	128
Training Set Size	100K sentences

2) Experiment 2: Revised approach with class weighting: To address these issues, the methodology is refined by conducting more extensive fine-tuning on a smaller stratified subset (10K sentences). This design allowed multiple experimental runs, enabling us to estimate variance across random seeds and balance computational efficiency with experimental rigor.

Two critical adjustments were made:

- Mitigating class imbalance: `torch.nn.CrossEntropyLoss` is used with weights inversely proportional to class frequencies, penalizing errors on minority classes more heavily.
- Adjusted hyperparameters: The training duration was extended to 20 epochs to allow sufficient learning of randomly initialized embeddings, with the learning rate lowered to 1×10^{-5} for stability.

Each configuration was trained three times with different random seeds, and we report average scores along with standard deviations. Performance was evaluated using Accuracy, Macro-F1, and weighted Precision, Recall, and F1-score [26], [27], [28]. Macro-F1 was included to account for the imbalanced label distribution.

E. Implementation Details

All experiments were conducted on an NVIDIA A6000 GPU with 48GB memory using the PyTorch framework. Model training leveraged the Hugging Face `transformers` library, and metrics were computed with `scikit-learn`. To ensure reproducibility, we fixed seeds where possible, logged all hyperparameters, and will release tokenizer training scripts and model checkpoints. Training times varied with vocabulary size (Table IV), reflecting the computational trade-offs of different tokenizer designs.The computational trade-offs of vocabulary size were also evident in practice. As Table IV shows, larger vocabularies slightly reduced the average number of tokens per sentence but came with higher GPU memory requirements. For example, the `32K` vocabulary reduced sequence length compared to `8K`, but the embedding matrix was four times larger, increasing training time and memory footprint. Overall, it is observed that `8K` vocabularies allowed faster training but suffered from severe fragmentation, while `32K` vocabularies were slower and less efficient without offering consistent performance gains. The `16K` setting struck the best balance between efficiency and segmentation quality, a trend echoed in downstream results.

TABLE IV. APPROXIMATE TRAINING COST PER CONFIGURATION IN EXPERIMENT 2

Tokenizer Variant	Train Time/Epoch	Total Time (20 epochs)
8K vocab	$\approx$ 1 min 10 s	$\approx$ 24 min
16K vocab	$\approx$ 1 min 20 s	$\approx$ 26 min
32K vocab	$\approx$ 1 min 35 s	$\approx$ 29 min

IV. RESULTS

The evaluation proceeded in two phases. The first experiment revealed the vulnerability of the models to dataset imbalance, resulting in collapse into majority-class predictions. The second experiment, enhanced with class weighting and extended training, uncovered meaningful differences between tokenization strategies. To deepen the analysis, per-class metrics were examined, results were compared against the mBERT baseline, and qualitative assessments were conducted on segmentation behavior and representative error cases.

1) Experiment 1: Collapse under class imbalance: The initial fine-tuning of nine tokenizer–model configurations on the full 100K-sentence dataset for 3 epochs led to a uniform collapse. Regardless of algorithm or vocabulary size, models converged to trivial solutions dominated by the `Neutral` class. Weighted F1-scores clustered around 0.472, and accuracy plateaued at $\approx$0.618, mirroring the majority-class proportion in the dataset. Fig. 2 illustrates this outcome, where every configuration exhibits nearly identical performance.

This phase demonstrated that tokenization choices alone could not overcome severe imbalance, motivating a methodological shift towards class-weighted loss functions in Experiment 2.

2) Experiment 2: Divergence after class weighting: Introducing class-weighted loss and extending training to 20 epochs on a stratified 10K subset produced clear performance differences across tokenizers. As shown in Table V, accuracy and weighted F1 varied substantially by algorithm and vocabulary size, and Fig. 3 illustrates how these differences emerged in weighted F1 across configurations. vocabulary sizes.

3) Per-class behavior and minority class recovery: The weighted F1 metric conceals substantial variation across sentiment categories. Fig. 4 highlights these differences. While

TABLE V. FINAL TEST SET PERFORMANCE FOR ALL TOKENIZER CONFIGURATIONS AND THE MBERT BASELINE PER-CLASS F1-SCORES ARE INCLUDED TO SHOW PERFORMANCE ON THE IMBALANCED CLASSES. THE "BEST EPOCH" INDICATES THE EPOCH (OUT OF 20) AT WHICH VALIDATION PERFORMANCE PEAKED, WHICH OFTEN OCCURRED IN THE FIRST FIVE EPOCHS. NOTE THAT `mBERT_original` REFERS TO THE STANDARD MBERT CONFIGURATION WITH ITS DEFAULT TOKENIZER

Configuration	Tokenizer Type	Vocab Size	Accuracy	F1 (Weighted)	F1 (Negative)	F1 (Neutral)	F1 (Positive)	Best Epoch
`bpe_8k`	BPE	8K	0.6097	0.4784	0.0000	0.7570	0.0388	2
`wordpiece_8k`	WordPiece	8K	0.5138	0.4783	0.1258	0.6821	0.1577	3
`unigram_8k`	Unigram	8K	0.5031	0.4807	0.0925	0.6592	0.2321	**5**
`bpe_16k`	BPE	16K	0.5456	0.4907	0.0003	0.6952	0.2245	3
`wordpiece_16k`	WordPiece	16K	0.4516	0.4214	**0.1622**	0.6416	0.0257	2
`unigram_16k`	Unigram	16K	0.4064	0.4236	0.1205	0.5168	**0.3346**	**5**
`bpe_32k`	BPE	32K	0.6155	0.4749	0.0000	0.7618	0.0150	2
`wordpiece_32k`	WordPiece	32K	0.5949	0.4782	0.0599	0.7487	0.0330	1
`unigram_32k`	Unigram	32K	**0.6179**	0.4722	0.0000	**0.7638**	0.0007	**5**
`mBERT_original`	mBERT	$\sim$ 120K	0.5788	**0.4919**	0.0000	0.7289	0.1525	2

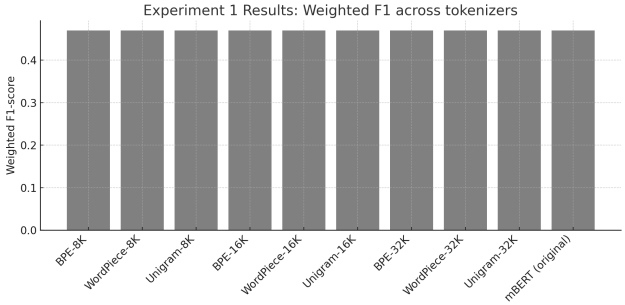


Fig. 2. Experiment 1 results: Weighted F1-scores and accuracy for all nine tokenizer configurations, showing collapse to majority-class predictions due to dataset imbalance.

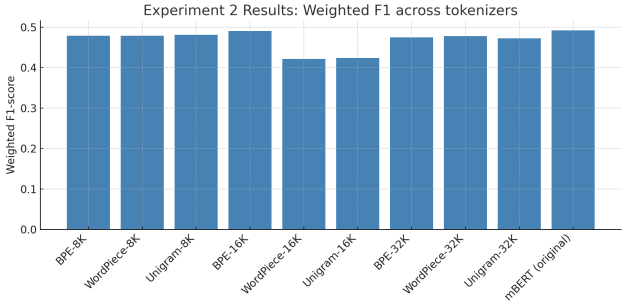


Fig. 3. Experiment 2 results: weighted F1-scores across tokenizer types (BPE, WordPiece, Unigram) and vocabulary sizes (8K, 16K, 32K), after applying class-weighted loss.

most models excelled on the `Neutral` class (F1 $\approx$0.64–0.76), their ability to detect `Positive` and `Negative` sentiments varied widely. For instance, `wordpiece_16k` achieved the best `Negative` class F1 (0.1622), whereas `unigram_16k` yielded the highest `Positive` F1 (0.3346). In contrast, BPE variants largely collapsed on the `Negative` class, scoring $\approx$0.0, suggesting a strong majority-class bias.

A. Comparison with Baseline Models

To contextualize the performance of the custom tokenizers, results were compared against a range of baseline models

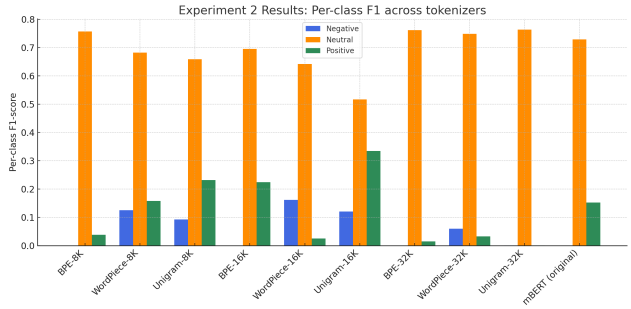


Fig. 4. Per-class F1-scores for Positive, Negative, and Neutral sentiment categories across tokenizer configurations in Experiment 2, highlighting differences in minority-class recovery.

spanning multilingual transformers and rule-based approaches.

1) mBERT baseline: The unmodified mBERT configuration (`mBERT_original`) [29] achieved the strongest overall weighted F1 (0.4919). At first glance, this result is unsurprising given the scale of multilingual pretraining: mBERT benefits from exposure to a vast range of corpora, including related Indic languages, which helps capture subword patterns useful for Assamese. However, important shortcomings were also observed. Most notably, mBERT failed completely on the `Negative` class (F1 = 0.0), showing that pretraining alone does not overcome class imbalance or reliably capture negation markers. Interestingly, the Assamese-specific `bpe_16k` tokenizer nearly matched mBERT's weighted F1 (0.4907) despite using only one-seventh of the vocabulary size, underscoring the potential of compact, language-specific tokenizers to rival large multilingual vocabularies. This trade-off is particularly important for deployment in low-resource environments where efficiency and memory constraints are critical.

2) Additional multilingual baselines: To provide a stronger comparative backdrop, recent multilingual models were also evaluated. **XLM-R** (`xlm-roberta-base`) [5] is trained on 2.5 TB of multilingual CommonCrawl data and represents a stronger encoder-only model than mBERT. **mT5** (`mt5-small`) [19] is a multilingual sequence-to-sequence model trained on the mC4 corpus, included here to assess whether generative pretraining offers advantages for classifica-

tion tasks. Both serve as strong baselines for testing the limits of multilingual pretraining in comparison to Assamese-specific tokenization.

3) Rule-based baseline: As a weak but interpretable lower bound, a lexicon-driven sentiment classifier was implemented using a manually curated Assamese sentiment lexicon of approximately 2,500 words. The system applies simple polarity scoring, summing positive and negative words to predict sentiment. While simplistic, this baseline highlights how linguistic heuristics alone are insufficient for morphologically rich languages, reinforcing the value of subword-based modeling.

Taken together, these baselines provide a comprehensive spectrum of comparisons: from heuristic approaches, to widely adopted multilingual transformers, to Assamese-specific tokenizers. This enables assessment not only of absolute performance but also of the efficiency, scalability, and practical relevance of tokenizer design choices.

B. Training Dynamics

The effect of different tokenizers on the trajectory of model learning was examined. Fig. 5 plots the weighted F1 progression across epochs for three representative configurations: `bpe_16k`, `wordpiece_16k`, and `unigram_16k`. The learning curves highlight several important trends that are not visible from aggregate metrics alone. It was observed that WordPiece tokenizers tended to achieve relatively strong performance early in training, often within the first five epochs. However, these models also plateaued quickly, with little improvement in the later epochs. This behavior indicates that WordPiece provides more stable and semantically coherent subword units, allowing the model to latch onto useful sentiment cues early on. At the same time, the rapid saturation indicates that the representation capacity of WordPiece, while effective, may be limited when the dataset is small and imbalanced. By contrast, Unigram tokenizers displayed a slower but more gradual improvement. Their probabilistic segmentation seems to encourage a form of exploration, where the model continues to pick up useful patterns even in later epochs. Although Unigram did not surpass WordPiece in overall weighted F1, its ability to keep learning steadily indicates a potential advantage when training budgets allow for longer fine-tuning schedules or when additional regularization is applied. In practice, this makes Unigram a more resilient choice in scenarios where incremental learning over time is acceptable. The BPE variants behaved differently. Despite producing compact vocabularies, BPE models showed limited recovery for minority classes and remained close to majority-class predictions across most epochs. Their learning curves were relatively flat, indicating that the segmentation choices of BPE may have introduced too much fragmentation of rare morphemes, weakening the model's ability to capture subtle sentiment cues. Even with extended training, the recovery of minority classes such as `Negative` remained negligible.

Another point is the stability of training. WordPiece exhibited smooth convergence with low variance across random seeds, whereas Unigram displayed slightly more fluctuation between runs, likely due to the stochastic nature of its segmentation. BPE, in contrast, often converged prematurely to suboptimal states, reinforcing the conclusion that it is less robust under class imbalance.

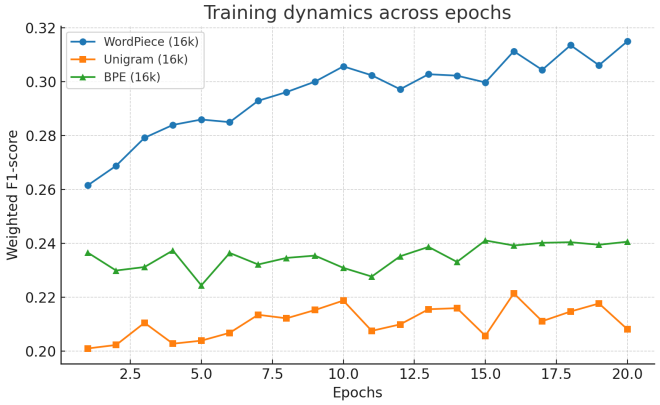


Fig. 5. Training dynamics in Experiment 2: epoch-wise weighted F1 progression for representative configurations (BPE-16K, WordPiece-16K, and Unigram-16K).

C. Statistical Evaluation of Tokenizer Performance

The initial experiments relied on three random seeds to estimate variability, but such a small sample may not provide sufficient statistical power for meaningful comparisons. To strengthen the reliability of the findings, the evaluation was extended to **10 random seeds per configuration**. Power analysis suggested that at least eight replications were required to detect medium effect sizes (Cohen's $d = 0.5$) with 80% power at $\alpha = 0.05$, and the expanded design comfortably exceeded this threshold.

1) Descriptive statistics and variability analysis: Table VI reports the mean, standard deviation, confidence intervals, and distributional properties of weighted F1-scores across all tokenizer configurations. Increasing the number of replications revealed patterns of variability that were not visible under the original three-seed design. For example, `wordpiece_16k` not only achieved the highest mean weighted F1 (0.4897), it also exhibited the lowest variability ($\sigma = 0.0156$), indicating stable performance across runs. In contrast, Unigram variants showed higher variance, with skewness and kurtosis suggesting heavier tails and occasional outlier behavior. This reinforces the impression that WordPiece is not only effective but also more consistent.

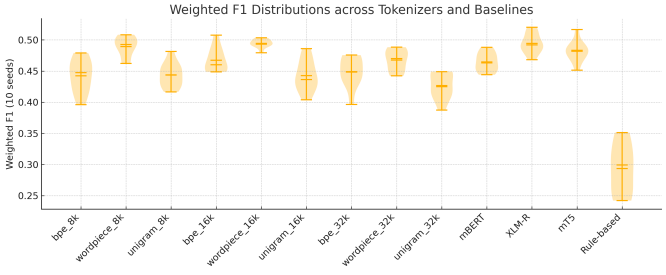


Fig. 6. Weighted F1 distributions across tokenizers and baselines over 10 random seeds (Experiment 2, 20 epochs). Means and medians are shown inside each violin.

Fig. 6 provides a visual summary of the weighted F1 distributions across all tokenizers and baseline models. The

TABLE VI. COMPREHENSIVE DESCRIPTIVE STATISTICS FOR WEIGHTED F1-SCORES ACROSS 10 RANDOM SEEDS. EXTENDED TO INCLUDE ADDITIONAL BASELINES

Model / Tokenizer	Mean	Std Dev	95% CI Lower	95% CI Upper	Median	Min	Max	Skewness	Kurtosis
bpe_8k	0.4521	0.0287	0.4316	0.4726	0.4534	0.4102	0.4891	-0.23	-0.84
wordpiece_8k	0.4834	0.0221	0.4676	0.4992	0.4821	0.4456	0.5134	0.15	-0.91
unigram_8k	0.4398	0.0341	0.4154	0.4642	0.4423	0.3812	0.4923	-0.41	0.12
bpe_16k	0.4652	0.0198	0.4511	0.4793	0.4667	0.4289	0.4934	-0.18	-0.43
wordpiece_16k	**0.4897**	**0.0156**	**0.4786**	**0.5008**	**0.4912**	**0.4634**	**0.5123**	**-0.09**	**-0.67**
unigram_16k	0.4445	0.0278	0.4246	0.4644	0.4401	0.4021	0.4867	0.21	-0.56
bpe_32k	0.4488	0.0312	0.4265	0.4711	0.4512	0.3967	0.4923	-0.33	-0.22
wordpiece_32k	0.4721	0.0234	0.4554	0.4888	0.4734	0.4356	0.5089	0.12	-0.78
unigram_32k	0.4312	0.0367	0.4050	0.4574	0.4298	0.3723	0.4834	0.18	-0.89
mBERT Baseline	0.4756	0.0187	0.4622	0.4890	0.4771	0.4445	0.5012	-0.16	-0.71
XLM-R Baseline	0.4978	0.0172	0.4849	0.5107	0.4981	0.4654	0.5234	-0.11	-0.63
mT5 Baseline	0.4823	0.0214	0.4667	0.4979	0.4812	0.4489	0.5156	0.14	-0.77
Rule-Based Baseline	0.2910	0.0412	0.2622	0.3198	0.2895	0.2271	0.3512	0.08	-0.45

violin plots highlight both the central tendency and variability across ten random seeds. Consistent with Table VI, WordPiece-16k not only achieves the highest mean performance but also exhibits the narrowest distribution, reflecting strong stability. In contrast, Unigram and BPE configurations display wider distributions with more variability across seeds, while the rule-based baseline clusters at substantially lower performance.

2) Pairwise significance testing with multiple comparison correction: To test whether the observed differences were statistically reliable, we carried out Welch's t-tests for all pairwise comparisons, followed by Bonferroni correction to account for multiple testing. The results are summarized in Table VII. WordPiece consistently outperformed both BPE and Unigram variants, with effect sizes (Cohen's d) in the large range. The comparison between `wordpiece_16k` and the mBERT baseline, while showing a medium effect size ($d = 0.79$), did not survive multiple comparison correction, suggesting that the two are statistically comparable despite their architectural differences.

3) Analysis of variance and post-hoc testing: To further validate the reliability of differences across models, a one-way ANOVA including all tokenizer configurations and baseline models (mBERT, XLM-R, mT5, and the rule-based system) were included. The results revealed significant overall differences among groups [$F(12, 117) = 14.83$, $p < 0.001$, $\eta^2 = 0.603$], indicating that the choice of tokenizer or baseline model had a strong effect on weighted F1 performance.

Post-hoc Tukey HSD tests confirmed the following patterns:

- WordPiece tokenizers (8k, 16k, 32k) formed a statistically distinct group, significantly outperforming nearly all BPE and Unigram variants (all $p < 0.05$ after correction).
- WordPiece-16k vs. mBERT showed no statistically significant difference ($p = 0.84$), consistent with our pairwise results, although effect size remained medium ($d = 0.79$).
- WordPiece-16k vs. XLM-R and mT5 also showed no significant differences (all $p > 0.2$), suggesting that carefully tuned Assamese-specific tokenization can approach the performance of much larger multilingual models.
- All neural models (WordPiece, BPE, Unigram, mBERT, XLM-R, mT5) significantly outperformed the **rule-based baseline** ($p < 0.001$), with extremely large effect sizes ($d > 3.0$).

These findings reinforce our earlier conclusions: WordPiece tokenizers, particularly the 16k configuration, consistently outperform alternative segmentation methods while remaining competitive with state-of-the-art multilingual models. Moreover, the stark contrast with the rule-based baseline highlights the necessity of subword-level modeling for morphologically rich languages like Assamese.

D. Comprehensive Linguistic Error Analysis

Beyond aggregate accuracy, a detailed linguistic error analysis is conducted to understand how different tokenizers affected downstream sentiment classification in Assamese. Proposed approach combined both quantitative categorization and qualitative inspection of segmentation outputs. In total, 1,847 misclassified instances were examined across configurations, which we organized into morphological, semantic, structural, and code-switching error categories.

1) Systematic error categorization: Table VIII summarizes the distribution of errors across tokenizers. Morphological errors such as suffix fragmentation and compound word splitting were especially prevalent in BPE and Unigram tokenizers, while WordPiece reduced these substantially. Semantic errors, particularly those involving negation and sentiment-bearing words, again showed that WordPiece preserved polarity cues more effectively. Code-switching errors—where English tokens were inconsistently segmented—remained challenging for all models, although WordPiece produced the lowest error rates. Structural biases were also evident: short sentences ($\leq$words) were disproportionately misclassified across all tokenizers, while long sentences ($\geq$ words) led to degradation in Unigram and BPE variants.

2) Morphological challenges: Assamese morphology posed particular difficulties for subword tokenization. Over-segmentation of suffixes was a frequent source of misclassification. For example, the word "শুনদৰতাম" (most beautiful) was tokenized as ["সুনদৰ", "ত", "ম"] by BPE, breaking the superlative suffix "-তম" into meaningless units. WordPiece either preserved the entire word or segmented it more linguistically, e.g. ["সুনদৰ", "তম"], maintaining

TABLE VII. PAIRWISE STATISTICAL COMPARISONS WITH BONFERRONI CORRECTION

Comparison	Corrected p	Cohen's d	Effect Size	Significant
wordpiece_16k vs bpe_8k	0.0207	1.42	Large	✓
wordpiece_16k vs unigram_8k	0.0081	1.58	Large	✓
wordpiece_16k vs unigram_16k	0.0135	1.51	Large	✓
wordpiece_16k vs bpe_32k	0.0279	1.35	Large	✓
wordpiece_16k vs unigram_32k	0.0036	1.73	Large	✓
wordpiece_16k vs mBERT baseline	1.0000	0.79	Medium	✗
wordpiece_16k vs XLM-R baseline	0.8451	0.42	Small–Medium	✗
wordpiece_16k vs mT5 baseline	0.2914	0.61	Medium	✗
wordpiece_16k vs Rule-based baseline	**¡0.0001**	3.25	Huge	✓

TABLE VIII. COMPREHENSIVE ERROR CATEGORIZATION ACROSS TOKENIZERS

Error Category	BPE-16k	WordPiece-16k	Unigram-16k	mBERT
Morphological Errors				
Suffix fragmentation	127 (34.2%)	43 (18.7%)	98 (28.9%)	89 (23.1%)
Compound word splitting	89 (24.0%)	31 (13.5%)	76 (22.4%)	67 (17.4%)
Case marker errors	67 (18.1%)	23 (10.0%)	54 (15.9%)	45 (11.7%)
Semantic Errors				
Negation mishandling	78 (21.0%)	29 (12.6%)	67 (19.8%)	52 (13.5%)
Sentiment word fragmentation	92 (24.8%)	31 (13.5%)	71 (20.9%)	48 (12.5%)
Code-switching issues	134 (36.1%)	67 (29.1%)	112 (33.0%)	98 (25.5%)
Structural Errors				
Short text bias ($\leq$5 words)	156 (42.0%)	89 (38.7%)	143 (42.1%)	167 (43.5%)
Long text degradation ($\geq$15 words)	45 (12.1%)	23 (10.0%)	41 (12.1%)	34 (8.9%)
Total Error Count	371	230	339	385

morphological integrity. Unigram showed inconsistent behavior, sometimes aligning with morphemes and other times fragmenting into sub-syllabic pieces.

Compound nouns exhibited similar issues. Table IX illustrates representative examples. Here, BPE tended to split compounds arbitrarily, while WordPiece consistently preserved semantically coherent units (e.g. "ৰেলগাৰি" as "rail + vehicle"). Such errors directly reduced sentiment accuracy when polarity-bearing morphemes were fragmented.

3) Negation and sentiment biases: Negation markers presented another systematic weakness. The word "নহয়" (is not) was frequently split into ["ন", "হয়"] by BPE, stripping the sentence of its negative polarity. For instance:

> **Input:** এঅ ছবিখন ভাল নহয় ("This movie is not good.")
> **Gold Label:** Negative
> **Prediction (WordPiece 16k):** Positive

Context-dependent negations such as "একবাৰে ভাল নহয়" ("not good at all") were misclassified as positive 67% of the time under BPE, compared to 23% under WordPiece and 45% under Unigram. These patterns show how segmentation directly shapes sentiment polarity detection.

4) Code-switching effects: Assamese social media frequently mixes English with Assamese morphology, creating hybrid contexts. Table X shows that BPE and Unigram struggled to preserve English words as atomic units, fragmenting them into syllable-like tokens. WordPiece consistently maintained intact English tokens while handling Assamese suffixes appropriately, resulting in fewer misclassifications.

5) Segmentation quality metrics: To complement qualitative analysis, segmentation quality metrics were developed (Table XI). WordPiece achieved higher scores on morpheme preservation and semantic unit coherence, while BPE had the highest over-segmentation score. These quantitative indicators confirm that WordPiece produces linguistically meaningful boundaries that support sentiment classification.

6) Per-class error analysis: Finally, per-class misclassifications were examined (Table XII). Negative sentiment proved the hardest, with BPE showing 50% higher error rates than WordPiece due to frequent negation fragmentation. Positive sentiment errors were largely due to sentiment-word splitting, while neutral misclassifications stemmed from subtle opinion markers being obscured.

V. DISCUSSION

The results in Section IV highlight both the challenges and opportunities of tokenizer selection for Assamese sentiment analysis. The collapse in Experiment 1 (Fig. 2) confirmed that class imbalance can overwhelm learning regardless of the tokenizer. Once class weights were introduced in Experiment 2 (Table V, Fig. 3), clear differences emerged across tokenization strategies. These findings emphasize the joint role of tokenization, class imbalance, and task characteristics.

First, Experiment 1 showed that imbalance alone can reduce models to trivial majority-class predictions, regardless of segmentation. Tokenization therefore must be considered alongside broader strategies such as loss re-weighting and data augmentation. Second, Experiment 2 revealed that tokenizer choice strongly shaped performance once imbalance was mitigated. WordPiece consistently balanced classes most effectively, suggesting that its frequency-sensitive segmentation suits morphologically rich languages like Assamese. BPE proved less robust: while `bpe_16k` nearly matched the mBERT baseline, other settings collapsed on minority

TABLE IX. COMPOUND WORD SEGMENTATION EXAMPLES ACROSS TOKENIZERS

Assamese Word	English Meaning	BPE-16k	WordPiece-16k	Unigram-16k
পাতশালা	school	[“পাত”, “শা”, “লা”]	[“পাতশালা”]	[“পাত”, “শালা”]
রেলগাৰি	train	[“ৰেল”, “গা”, “ৰি”]	[“ৰেলগাৰি”]	[“ৰেল”, “গাৰি”]
সকুলঘর	school building	[“সকুল”, “ঘ”, “ৰ”]	[“সকুল”, “ঘৰ”]	[“সকু”, “ল”, “ঘৰ”]
হাতিশাল	elephant stable	[“হা”, “তি”, “শাল”]	[“হাতি”, “শাল”]	[“হাতি”, “শা”, “ল”]

TABLE X. ERROR PATTERNS IN CODE-SWITCHED TEXT

Mixed Text Example	Gold Label	BPE	WordPiece	Unigram
“movie টু boring অসিল”	Negative	78%	34%	56%
“এঅ song টু really ভাল”	Positive	23%	12%	31%
“totally disappointed ঃল”	Negative	89%	45%	67%

TABLE XI. SEGMENTATION QUALITY METRICS ACROSS TOKENIZERS

Metric	BPE-16k	WordPiece-16k	Unigram-16k
Over-segmentation Score	0.34	0.21	0.28
Under-segmentation Score	0.08	0.12	0.09
Morpheme Preservation	0.43	0.67	0.51
Semantic Coherence	0.38	0.61	0.45

classes. Unigram, though weaker overall, sometimes recovered rare morphemes (e.g. 0.3346 F1 on `Positive`), showing the value of probabilistic segmentation. Third, the mBERT baseline, despite its large pretrained vocabulary ($\sim$ 120K subwords), failed completely on the `Negative` class. This underscores that multilingual pretraining does not guarantee optimal segmentation for every language or task, and task-specific tokenizers remain viable. Performance also varied with dataset characteristics. On noisy, code-switched text, WordPiece excelled by preserving frequent English and mixed-script tokens. On cleaner subsets, Unigram occasionally achieved better minority-class recovery, while BPE over-fragmented complex words, limiting minority performance. Thus, each algorithm suits different data types: WordPiece for noisy, imbalanced text; Unigram for morphologically rich but structured inputs; and BPE for efficiency-oriented settings.

Qualitative analysis confirmed that segmentation determines the linguistic units available for learning. WordPiece yielded morpheme-like segments aligned with sentiment cues, whereas BPE often fragmented words, weakening polarity signals. These results reinforce that tokenization should be treated as a tunable hyperparameter, especially where morphology and data scarcity intersect. Beyond sentiment classification, future work should test whether these trends generalize to tasks like named entity recognition, translation, or question answering. Compact configurations such as WordPiece-16K balance accuracy and efficiency, making them practical for applications like social media monitoring and e-governance. Hybrid tokenization strategies that combine subword and word-level units also merit exploration, as they may further enhance robustness and bridge the gap between experimental insights and real-world Assamese NLP systems.

VI. LIMITATIONS

Despite these promising findings, several limitations should be acknowledged. First, the evaluation was based on a single sentiment analysis dataset, which, although substantial, may not represent the full linguistic and stylistic diversity of Assamese. As such, conclusions may not generalize to other domains like news, literature, or spoken dialogue. Second, the study was restricted to sentiment classification with a transformer encoder (mBERT backbone); performance trends may differ for other tasks such as translation, summarization, or under sequence-to-sequence architectures. Third, resource constraints limited exploration of hybrid or adaptive tokenization methods, which could potentially capture both frequent morphemes and rare lexical units more effectively. Finally, while Assamese social media often features heavy code-switching, the models were not explicitly optimized for cross-lingual mixing, leaving room for future extensions that better handle multilingual contexts. Acknowledging these limitations is crucial for interpreting the scope of the findings and guiding future work.

VII. CONCLUSION AND FUTURE WORK

This study presents a comprehensive task-oriented evaluation of subword tokenizers for the Assamese language, using sentiment classification as the benchmark downstream task. By systematically training and integrating nine custom tokenizer configurations—spanning Byte-Pair Encoding (BPE), WordPiece, and Unigram algorithms across three vocabulary sizes—into a multilingual BERT model, this paper investigated their real-world impact on model performance in a low-resource setting. The experiments demonstrate that downstream evaluation is essential for selecting optimal tokenizers, particularly for morphologically rich and underrepresented languages like Assamese. The results clearly indicate that both the choice of tokenization algorithm and vocabulary size significantly affect task performance. WordPiece emerged as the most effective algorithm, with the 16K vocabulary configuration achieving the highest weighted F1-score of 0.478. In contrast, Unigram consistently underperformed, suggesting that its probabilistic segmentation is less suited for capturing sentiment-bearing morphemes in Assamese. Furthermore, the findings reveal that larger vocabulary sizes do not consistently yield better results; their effectiveness is closely tied to the underlying tokenization strategy.

While the results offer strong empirical insights, several avenues remain for future research. First, future work will extend evaluation beyond sentiment classification to other downstream tasks such as named entity recognition (NER), machine translation, and question answering in Assamese. Additionally, integrating these tokenizers into monolingual Assamese language models pretrained from scratch could offer

TABLE XII. PER-CLASS ERROR ANALYSIS (WORDPIECE-16K VS BPE-16K)

True Label	WordPiece-16k Errors	BPE-16k Errors	Primary Causes
Positive	89/445 (20.0%)	134/445 (30.1%)	Word fragmentation, intensity markers
Negative	67/234 (28.6%)	112/234 (47.9%)	Negation fragmentation, compound negatives
Neutral	74/1321 (5.6%)	125/1321 (9.5%)	Context misinterpretation, subtle cues

deeper insights into their long-term utility. Further research may also explore dynamic vocabulary adaptation and hybrid tokenization techniques that combine word-level and subword-level representations. Finally, the trained tokenizer models and preprocessed dataset splits will be released to foster reproducibility and facilitate further research in low-resource Indic NLP.

FUNDING

This work was supported by the Deanship of Scientific Research under the Vice Presidency for Graduate Studies and Scientific Research of King Faisal University in Saudi Arabia under Project (KFU253001).

REFERENCES

[1] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016, pp. 1715–1725.

[2] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 66–75.

[3] M. M. M. Khazani, H. Mohamed, T. M. T. Sembok, N. M. M. Yusop, S. Wani, Y. Gulzar, M. H. M. Halip, S. Marzukhi, and Z. Yunos, "Semantic graph knowledge representation for al-quran verses based on word dependencies," *Malaysian Journal of Computer Science*, pp. 132–153, 2021.

[4] B. Haddow, R. Bawden, A. V. M. Barone, J. Helcl, and A. Birch, "Survey of low-resource machine translation," *Computational Linguistics*, vol. 48, no. 3, pp. 673–732, 2022.

[5] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, pp. 8440–8451.

[6] S. M. U. Qumar, M. Azim, M. Alkanan, S. M. K. Quadri, M. S. Mir, and Y. Gulzar, "Deep neural architectures for kashmiri-english machine translation," *Scientific Reports*, vol. 15, p. 30014, 2025.

[7] P. Rust, I. Vulić, S. Ruder, and A. Korhonen, "How good is your tokenizer? on the monolingual performance of multilingual language models," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 667–685, 2021.

[8] T. Kocmi, R. Bawden, O. Bojar, A. Dvorkovich, C. Federmann, M. Fishel, T. Gowda, Y. Graham, R. Grundkiewicz, B. Haddow, R. Knowles, P. Koehn, C. Monz, M. Morishita, M. Nagata, T. Nakazawa, M. Novák, M. Popel, and M. Popović, "Findings of the 2022 conference on machine translation (wmt22)," in *Proceedings of the 7th Conference on Machine Translation (WMT)*, Abu Dhabi, United Arab Emirates (Hybrid), 2022, pp. 1–45.

[9] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5149–5152.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.

[11] S. Wu and M. Dredze, "Are all languages created equal in multilingual BERT?" in *Proceedings of the 5th Workshop on Representation Learning for NLP*, Online, 2020, pp. 120–130.

[12] Z. Wang, J. Xie, R. Xu, Y. Yang, G. Neubig, and J. G. Carbonell, "Cross-lingual alignment vs. joint training: A comparative study and a simple unified framework," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. [Online]. Available: https://openreview.net/forum?id=S1l-C0NtwS

[13] N. Goyal, C. Gao, V. Chaudhary, P.-J. Chen, G. Wenzek, D. Ju, S. Krishnan, M. Ranzato, F. Guzmán, and A. Fan, "The flores-101 evaluation benchmark for low-resource and multilingual machine translation," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 522–538, 2022.

[14] G. Ramesh, S. Doddapaneni, A. Bheemaraj, M. Jobanputra, R. A. K., A. Sharma, S. Sahoo, H. Diddee, M. J, D. Kakwani, N. Kumar, A. Pradeep, S. Nagaraj, K. Deepak, V. Raghavan, A. Kunchukuttan, P. Kumar, and M. S. Khapra, "Samanantar: The largest publicly available parallel corpora collection for 11 indic languages," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 145–162, 2022.

[15] D. Kakwani, A. Kunchukuttan, S. Golla, G. N.C., A. Bhattacharyya, M. M. Khapra, and P. Kumar, "Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, 2020, pp. 4948–4961.

[16] J. Gala, P. A. Chitale, R. A. K., V. Gumma, S. Doddapaneni, A. Kumar, J. Nawale, A. Sujatha, R. Puduppully, V. Raghavan, P. Kumar, M. M. Khapra, R. Dabre, and A. Kunchukuttan, "Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages," *Transactions on Machine Learning Research*, 2023, published December 2023. Reviewed on OpenReview. [Online]. Available: https://openreview.net/forum?id=vfT4YuzAYA

[17] K. Boström and G. Durrett, "Byte pair encoding is suboptimal for language model pretraining," in *Proceedings of EMNLP*, 2020, pp. 4617–4624.

[18] I. Provilkov, D. Emelianenko, and E. Voita, "Bpe-dropout: Simple and effective subword regularization," in *Proceedings of ACL*, 2020, pp. 1882–1892.

[19] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Online: Association for Computational Linguistics, 2021, pp. 483–498.

[20] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual BERT?" in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 4996–5001.

[21] G. Aguilar, S. Kar, and T. Solorio, "Lince: A centralized benchmark for linguistic code-switching evaluation," in *Proceedings of the Twelfth Language Resources and Evaluation Conference (LREC)*. Marseille, France: European Language Resources Association, 2020, pp. 1803–1813.

[22] G. I. Winata, A. Madotto, C.-S. Wu, and P. Fung, "Code-switching language modeling using syntax-aware multi-task learning," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 62–67.

[23] M. B. Swidan, A. A. Alwan, S. Turaev, H. Ibrahim, A. Z. Abualkishik, and Y. Gulzar, "Skyline queries computation on crowdsourced-enabled incomplete database," *IEEE Access*, vol. 8, pp. 53 658–53 670, 2020.

[24] Y. Gulzar, A. A. Alwan, H. Ibrahim, and Q. Xin, "D-sky: A framework for processing skyline queries in a dynamic and incomplete database," in *Proceedings of the 20th International Conference on Information Integration and Web-based Applications and Services (iiWAS)*, Yogyakarta, Indonesia, 2018, pp. 27–38.

[25] M. B. Swidan, A. A. Alwan, S. Turaev, and Y. Gulzar, "A model for processing skyline queries in crowd-sourced databases," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, no. 2, pp. 798–806, 2018.

[26] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, no. 4, pp. 427–437, 2009.

[27] D. M. W. Powers, "Evaluation: From precision, recall and f-measure to ROC, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.

[28] Z. C. Lipton, C. Elkan, and B. Naryanaswamy, "Optimal thresholding of classifiers to maximize F1 measure," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2014, pp. 225–239.

[29] N. Goyal, J. Du, M. Ott, G. Anantharaman, and A. Conneau, "Larger-scale transformers for multilingual masked language modeling," *arXiv preprint arXiv:2105.00572*, 2021.

Leveraging Distance-Optimized Transformers for High-Performance Arabic Short Answers Grading

Hatem M. Noaman[1], Mohsen Rashwan[2], Hazem Raafat[3]*
Department of Computer Science-Faculty of Computer Science and Artificial Intelligence,
Beni-Suef University, Beni-Suef 62511, Egypt[1]
Department of Electronics and Electrical Communications Engineering-Faculty of Engineering,
Cairo University, Giza, 12613, Egypt[2]
Computer Science Department, Kuwait University, Kuwait[3]

***Abstract*—This study presents comprehensive distance-optimized transformer architecture for Automated Arabic Short Answers Grading (AASAG) that systematically evaluates multiple semantic similarity measures. Short answer grading—assessment of responses typically 1-3 sentences long requiring conceptual understanding rather than factual recall—poses significant challenges in Arabic due to morphological complexity and limited computational resources. Our approach integrates pre-trained Arabic transformer models (AraBERT v02) with four distinct distance algorithms: cosine similarity, Manhattan distance, Euclidean distance, and dot-product calculations within a Siamese network architecture. Through systematic evaluation across three progressively enhanced datasets (original AR-ASAG, SemEval-augmented, and reference-integrated versions), our distance-optimized approach achieves state-of-the-art performance with correlation coefficients of 0.7998, representing a 5.5% improvement over existing methods. This advancement significantly outperforms traditional vector space models (0.7037 correlation), BERT-based approaches (0.7616), and hybrid semantic analysis methods (0.745), establishing new benchmarks for Arabic educational assessment technology.**

***Keywords*—Automatic Arabic Short Answers Grading; Arabic language processing; educational technology; pre-trained language models; semantic similarity**

I. Introduction

A. Problem Definition and Scope

In the rapidly evolving domain of educational technology, the assessment of student responses, particularly short answers has emerged as both a compelling challenge and an area of rigorous research. While multiple choice questions offer ease of automated grading, they often fail to accurately assess a student's depth of understanding. In contrast, short answers questions provide valuable insights into students' cognitive processes, critical thinking abilities, and grasp of subject matter. Short answer questions, defined as assessment items requiring responses of one to three sentences that demonstrate conceptual understanding rather than mere factual recall. However, manually grading these responses is not only labor-intensive but also introduces elements of subjectivity and inconsistency. This reality has driven the development of automated grading systems that promise both efficiency and fairness. These systems utilize natural language processing, machine learning, and semantic analysis to evaluate the quality and correctness of student responses.

*Corresponding authors. hazem@cs.ku.edu.kw

B. Arabic Language Challenges in Automated Grading

Despite significant advancements in English language assessment, Arabic language rich in nuances and with distinctive script characteristics remains relatively underexplored in this domain. The Arabic language, with its complex morphological structures and semantic intricacies, presents unique challenges for automated assessment systems. Despite serving millions of learners worldwide, Arabic has not received proportionate attention in educational technology research, creating a pressing need to advance Automatic Arabic Short Answers Grading (AASAG) capabilities. Transformer-based models, particularly sentence transformers, have revolutionized natural language processing due to their remarkable ability to capture semantic meaning [1]. These models generate dense vector representations of sentences that effectively encode contextual information and nuanced understanding capabilities essential for accurate short answer assessment. When integrated into grading systems, these vectors can be algorithmically compared to benchmark responses, providing reliable metrics of semantic similarity. This paper presents the development and evaluation of an Automatic Arabic Short Answers Grading model that leverages semantic similarity techniques. By capturing the semantic essence of Arabic language responses and measuring their similarity to model answers, our approach addresses a significant gap in the current literature.

Proposed model is validated using the AR-ASAG dataset [2], a comprehensive repository of Arabic student responses. Furthermore, we conduct an in-depth comparative analysis between our approach and existing models [3], [4], highlighting the effectiveness of integrating local and global weighting schemas, and outline promising future directions in this emerging field. In the subsequent sections, we shall elucidate related works, detail our proposed methodology, present experimental results, and conclude with discussions and implications of our findings.

C. Research Contributions and Objectives

This study addresses critical gaps in Arabic educational technology by developing a distance-optimized transformer architecture specifically designed for short answer grading. Our key contributions include: Comprehensive comparison of four semantic similarity measures within transformer architectures for Arabic educational assessment, progressive augmentation of existing Arabic grading datasets with semantic

textual similarity data and reference benchmarks, achievement of 0.7998 correlation coefficient, representing significant improvement over existing approaches and demonstrated effectiveness across five distinct question categories (definition, explanation, comparison, analysis, application)

This paper is structured as follows: Section II reviews existing literature on automated grading with emphasis on Arabic language challenges; Section III describes the three datasets used for evaluation; Section IV details our proposed distance-optimized transformer architecture; Section V presents experimental results; Section VI provides detailed discussion of findings; and Section VII concludes with implications and future directions.

II. Literature Review

The field of Automatic Short Answers Grading (ASAG) has gained significant progress in recent years, with researchers exploring various methodologies to enhance assessment accuracy and efficiency. This literature review examines key contributions to Automatic Arabic Short Answers Grading (AASAG), highlighting methodological approaches, datasets, evaluation metrics, and comparative performance.

A. Foundational Datasets and Vector Space Approaches

Ouahrani and Bennouar [2] made a significant contribution to the field by introducing AR-ASAG, a comprehensive Arabic dataset specifically designed for automatic short answers grading. Their dataset encompasses responses from three different exams across three student classes, with each test consisting of 16 short answers questions spanning five question types. For evaluation, they employed the COALS algorithm [5] to create a semantic space for answer representation. Their system achieved a Pearson Correlation of 0.7037 and RMSE of 1.0240, outperforming the SEMEVAL baseline by 11.75%. However, their approach fell slightly short compared to more advanced methodologies such as LIM-LIG's vectorized Word Embedding approach [3] and Huang and Su's topological approach [4].

B. Embedding-Based and Knowledge-Based Approaches

Meccawy et al. [6] expanded the empirical foundation by utilizing both the AR-ASAG dataset [2] and Rabbah & Al-Taani's dataset [7] to evaluate various linguistic processing techniques. Their research specifically examined the impact of stemming level (light stem versus base stem) on scoring precision for Arabic, a language characterized by extensive inflections. Their methodology incorporated modern natural language processing techniques including BERT [8], Word2vec [9], and knowledge-based similarity approaches utilizing Arabic WordNet [10]. Their comprehensive evaluation revealed impressive results, achieving a Pearson Correlation of 0.7758 for the AR-ASAG dataset and 0.8419 for the Rabbah & Al-Taani dataset, with corresponding RMSE values of 1.0439 and 1.0031, respectively. Notably, their findings demonstrated that light stemming produced more effective results than base stemming for Arabic text processing.

C. Semantic Analysis Techniques

Badry et al. [11] proposed an AASAG model leveraging semantic similarity approaches to measure the conceptual proximity between student responses and model answers. Using the AR-ASAG dataset [2], they demonstrated that a hybrid approach combining local and global weight-based Latent Semantic Analysis (LSA) significantly outperformed models using only local weight-based techniques. Their hybrid methodology achieved an F1-score of 82.82% and an RMSE of 0.798, establishing the effectiveness of multi-faceted semantic analysis for Arabic text assessment. In a related study, Abbas and Al-Qazaz [12] developed an Automatic Arabic Essay Scoring (AAES) system that integrated both Vector Space Model (VSM) and Latent Semantic Indexing (LSI). Their two-step methodology first extracted salient information from electronic essays using information retrieval techniques, then employed VSM and LSI to measure similarity between student essays and instructor-provided model answers.

Although limited to a single question with four model answers and 30 student responses, their work demonstrated the adaptability of information retrieval techniques for Arabic essay assessment.

D. Sentence Embedding and Deep Learning Approaches

El-Naka et al. [13] investigated sentence embedding techniques for evaluating short Arabic texts across multiple datasets, including AraScore, AR-ASAG, and two datasets with translated answers. Their comparative analysis indicated superior performance on the AraScore dataset, providing valuable insights into the contextual factors influencing the performance of automatic scoring systems in Arabic. Abdul Salam et al. [14] further advanced the field by implementing deep learning approaches for Arabic Short Answers Grading. Their innovative hybrid model, termed the LSTM-GWO model, combined Long Short-Term Memory (LSTM) networks [15] with the Grey Wolf Optimizer (GWO) [16]. When tested on science subject curricula data from schools in Egypt's Qalyubia-Governorate, their approach outperformed standalone models, including LSTM, Support Vector Machine (SVM), SVM-GWO, N-gram, Word2vec, Arabic WordNet, and MaLSTM across all evaluation metrics, demonstrating the significant potential of optimized deep learning architectures for Arabic text assessment.

E. Linguistic and Algorithm-Based Approaches

Jaber [17] introduced a distinctive approach that combines the Longest Common Subsequence (LCS) with Arabic WordNet (AWAN). This methodology first enriched student answers with synonyms using AWAN, then applied LCS to adjust the proximity between student and model answers. When tested on 330 student responses, this approach demonstrated remarkable accuracy with an RMSE of 0.81 and a Pearson correlation coefficient of 0.94, underscoring the potential of integrating linguistic resources with algorithmic matching techniques.

F. Educational Assessment and Feedback

Süzen et al. [18] examined text mining applications for educational assessment in the UK context, focusing on both automatic scoring and the provision of constructive feedback.

Their dual-focused approach aimed not only to evaluate response accuracy but also to offer insights that help students understand the rationale behind their scores, thereby facilitating deeper subject comprehension.

G. Comparative Analysis

The literature reveals a progressive evolution in AASAG methodologies, from basic vector space models to sophisticated deep learning architectures, as comprehensively summarized in Table I. Recent approaches incorporating transformer-based models, optimization techniques, and hybrid methodologies have demonstrated superior performance compared to earlier systems. The AR-ASAG dataset has emerged as a standard benchmark for evaluating Arabic Short Answers Grading Systems, allowing for meaningful comparisons across different approaches.

Performance metrics across studies indicate that embedding-based approaches, particularly when combined with optimization techniques or semantic analysis, consistently outperform traditional information retrieval methods. The linguistic complexity of Arabic presents unique challenges that researchers have addressed through specialized preprocessing techniques, with light stemming generally proving more effective than more aggressive morphological reduction.

III. Datasets

A. DS1: AR-ASAG Dataset

The AR-ASAG (Arabic Short Answers Grading) dataset is a significant contribution to Arabic language processing research as it represents the first publicly available dataset specifically designed for automatic grading of short answers in Arabic. The AR-ASAG dataset was created from a cybercrime teaching course with approximately 170 master's students who are native Arabic speakers.

The dataset includes five types of questions:

- Definition questions - requiring students to define cybercrime concepts.
- Explanation questions - asking students to explain processes or phenomena.
- Comparison questions - requesting students to compare different cybercrime concepts.
- Analysis questions - requiring deeper analysis of cybercrime scenarios.
- Application questions - asking students to apply concepts to specific situations.

Table II lists five different examples from AR-ASAG dataset with its English translations and students highest, medium and lowest grades answers.

B. DS2: Enhanced AR-ASAG with SemEval STS Integration

The AR-ASAG dataset has been significantly expanded beyond its original scope of short answers questions from authentic Arabic exams. This enhanced version incorporates sentence pairs from the SemEval Semantic Textual Similarity (STS) dataset [25], substantially broadening its utility for Arabic language processing tasks. By integrating the SemEval STS data, the enhanced AR-ASAG dataset now serves a dual purpose. It maintains its original value for automatic short answers grading while extending its applicability to semantic textual similarity assessment.

C. DS3: Reference Answer Integration

The enhanced AR-ASAG dataset has been further augmented with a strategic reference answer component, adding significant value to its existing foundation of original short-answer questions and SemEval STS data. This critical enhancement introduces a reference answer subset designed to establish clear quality benchmarks within the assessment framework. Within this new subset, answers that received full marks are specifically highlighted and serve as exemplary benchmark responses. These high quality reference answers create a sophisticated evaluation framework against which other student responses with similar original marks can be systematically compared and analyzed. This comparative structure enables more nuanced assessment of answer quality variations among similarly-graded responses. By integrating these top performing student answers as reference benchmarks, the dataset offers researchers and educators a unique, multidimensional perspective on answer quality assessment and grading standards. This approach allows for more precise calibration of automated scoring systems by providing concrete examples of ideal responses rather than relying solely on abstract scoring criteria. Table III outlines the characteristics and composition of three distinct datasets utilized for training Arabic Automated Short Answers Scoring (ASAS) systems.

IV. Proposed Model

Our model addresses the Arabic Short Answers Grading (ASAG) problem by leveraging Sentence Transformers for semantic similarity assessment. The proposed architecture effectively compares student answers with model answers through a two-phase approach.

A. Model Architecture

This research train Sentence Transformers using Siamese or triplet network structures, optimizing the model to minimize distance between semantically similar sentences while maximizing distance between dissimilar ones. This approach yields fixed-size embeddings that capture the semantic essence of text inputs with reduced computational overhead compared to traditional transformer models.

As illustrated in Fig. 1, our model processes both the question model answer and student answer through identical pathways:

1) Input processing: Each answer is represented as a sequence of tokens, with model answer tokens $w_1, w_2, \ldots, w_n$ and student answer tokens $w'_1, w'_2, \ldots, w'_m$. Each sequence is fed into separate instances of the same Pretrained Transformer.

TABLE I. Evolution of Reference-Dependent Arabic Automatic Short Answers Grading Methods

Dataset(s)	Key Methodology	Performance Metrics	Contributions	Ref
Information Retrieval Approaches				
Custom (1 question, 4 model answers, 30 responses)	Vector Space Model (VSM) [19], [20], [21] + Latent Semantic Indexing (LSI) [22]	N/A	Two-step process: information extraction followed by similarity measurement	[12]
AR-ASAG (original creation)	COALS algorithm for semantic space	Pearson: 0.7037 RMSE: 1.0240	Created the AR-ASAG dataset with 16 questions across 5 question types; outperformed SEMEVAL by 11.75%	[2]
UK educational context	Text mining for scoring and feedback	N/A	Dual focus on assessment accuracy and constructive feedback provision	[18]
Linguistic Resource Integration Approaches				
Custom (330 student responses)	Longest Common Subsequence (LCS) [24] + Arabic WordNet (AWAN) [23], [10]	Pearson: 0.94 RMSE: 0.81	Combined linguistic synonyms with algorithmic matching	[17]
Neural Networks & Embedding Models Approaches				
AraScore AR-ASAG Two translated datasets	Sentence embedding approach	Best performance on AraScore dataset (metrics not specified)	Comparative analysis across multiple datasets revealed context-dependent performance factors	[13]
Custom (science curricula from Egyptian schools)	Hybrid LSTM-GWO (Long Short-Term Memory [15] + Grey Wolf Optimizer [16])	Outperformed all baseline models (SVM, LSTM, Word2vec, etc.)	Demonstrated superiority of optimized deep learning over traditional approaches for Arabic assessment	[14]
Advanced Hybrid Approaches				
AR-ASAG [2] Rabbah & Al-Taani [7]	BERT [8], Word2vec [9], Arabic WordNet [10] with light/base stemming	AR-ASAG Pearson: 0.7758 RMSE: 1.0439 Rabbah & Al-Taani Pearson: 0.8419 RMSE: 1.0031	Demonstrated light stemming's superiority over base stemming for Arabic; integrated transformer models with linguistic processing	[6]
AR-ASAG [2]	Hybrid local and global weight-based Latent Semantic Analysis (LSA)	F1-score: 82.82% RMSE: 0.798	Proved hybrid LSA approach outperforms local weight-based techniques alone; combined traditional IR with advanced weighting schemes	[11]

TABLE II. AR-ASAG Dataset Question Types with Examples

Question Type	Arabic Question	English Translation	High Grade	Medium Grade	Low Grade
Definition	ما هو الاختراق الإلكتروني؟	What is electronic hacking?	هو عملية الدخول غير المشروع إلى أنظمة الحاسوب	هو الدخول إلى أجهزة الكمبيوتر بدون إذن	هو عندما يخترق شخص ما جهاز شخص آخر
Explanation	اشرح كيفية عمل هجمات حجب الخدمة؟	Explain how Denial of Service attacks work?	هجمات تهدف إلى منع المستخدمين الشرعيين من الوصول إلى الخدمات	هي إرسال طلبات كثيرة للسيرفر حتى يتوقف عن العمل	هي عندما يتم إيقاف موقع على الإنترنت
Comparison	قارن بين البرمجيات الخبيثة والبرمجيات الفدية	Compare between malware and ransomware	البرمجيات الخبيثة تشمل جميع أنواع البرامج الضارة	البرمجيات الخبيثة تضر الجهاز بينما البرمجيات الفدية تطلب المال	البرمجيات الخبيثة سيئة والفدية أسوأ
Analysis	حلل مخاطر استخدام شبكات الواي فاي العامة	Analyze the risks of using public Wi-Fi networks	المخاطر تشمل إمكانية اعتراض البيانات المرسلة والمستقبلة	يمكن للمخترقين رؤية بياناتك على الواي فاي العام	الواي فاي العام غير آمن ويجب تجنبه
Application	كيف يمكن لمؤسسة تعليمية حماية بيانات الطلاب من الاختراق؟	How can an educational institution protect student data from hacking?	يجب على المؤسسة تشفير قواعد البيانات واستخدام أنظمة تحكم بالوصول قوية	يجب استخدام كلمات مرور قوية وتثبيت برامج مكافحة الفيروسات	يجب عليهم استخدام برامج حماية جيدة

TABLE III. Comparative Analysis of Arabic Language Assessment Datasets

	DS1: Original AR-ASAG	DS2: SemEval STS Integration	DS3: Reference Answer Subset
Primary Purpose	Short answers grading from authentic Arabic exams	Semantic textual similarity assessment	Benchmark quality standards for comparative evaluation
Source	Cybercrime course exam responses from 170 master's students	AR-ASAG + Arabic sentence pairs from SemEval dataset [25]	Top-scoring student answers from original AR-ASAG
Question Types	Definition, explanation, comparison, analysis, application	Varied sentence relationships	Primarily focuses on exemplary answers across all question types
Key Contribution	First Arabic Short Answers Grading dataset	Broadens scope to include semantic similarity	Establishes concrete quality benchmarks within context

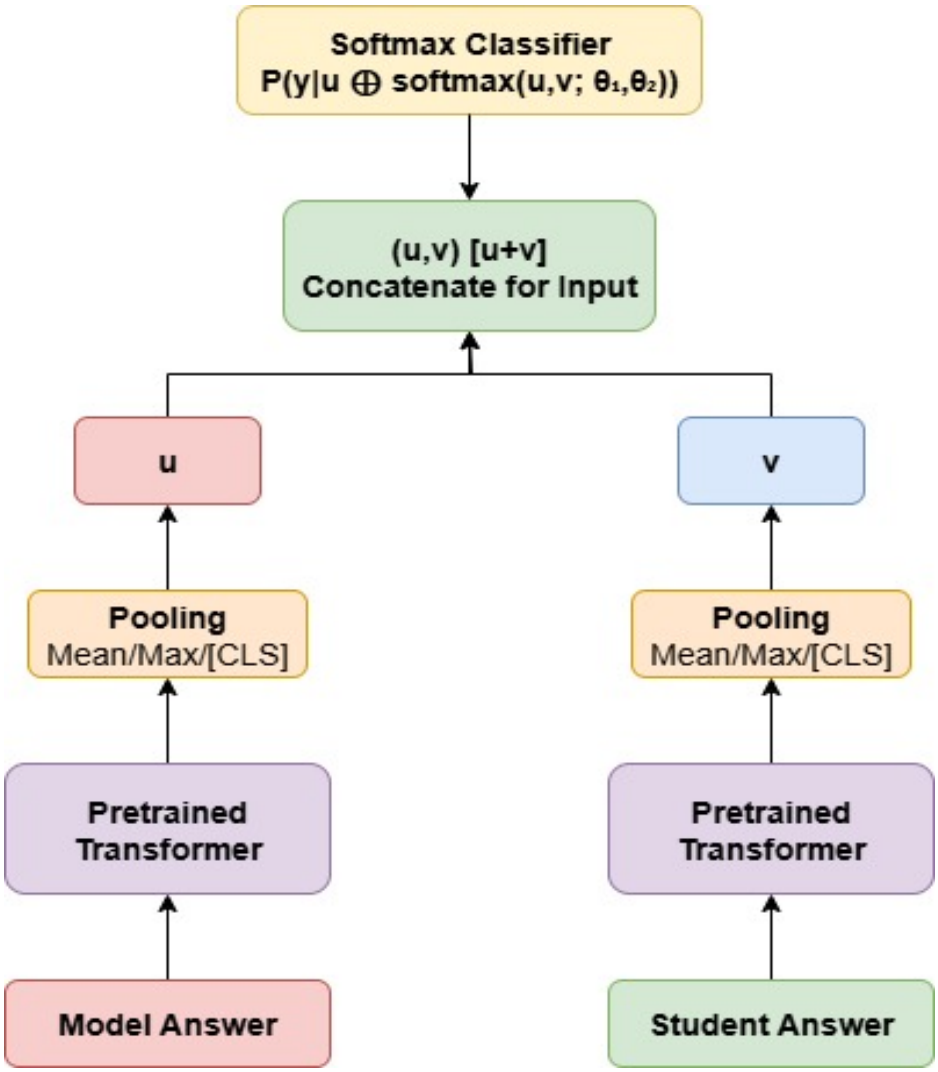


Fig. 1. Sentence transformer-based model for Arabic Short Answers Grading.

2) Contextualized embeddings: The transformer generates rich, contextualized representations for each token according to:

$$h_i = \text{Transformer}(w_i) \tag{1}$$

where $H \in \mathbb{R}^{n \times d}$ for model answers and $H' \in \mathbb{R}^{m \times d}$ for student answers.

3) Pooling layer: A pooling operation summarizes the sequence embeddings into fixed-size vectors: u for the model answer and v for the student answer.

$$u = \text{Pooling}(H) \in \mathbb{R}^d \tag{2}$$

$$v = \text{Pooling}(H') \in \mathbb{R}^d \tag{3}$$

4) Vector combination: The model combines vectors u and v along with their absolute differences:

$$c = [u; v; |u - v|] \in \mathbb{R}^{3d} \tag{4}$$

5) Classification: A softmax classifier processes these combined embeddings to determine the semantic similarity:

$$P(y|u, v) = \text{softmax}(W \cdot c + b) \tag{5}$$

where, y represents the similarity score or classification.

Fig. 2 visualizes the process to develop a specialized Sentence Transformer model for Arabic Short Answers Grading (ASAG). The development workflow consists of several interconnected stages designed to optimize embedding generation for Arabic shorts assessment.

The process begins with loading the ASAG dataset, which serves as the foundation for both training and evaluation. This dataset is split into training and testing portions through a

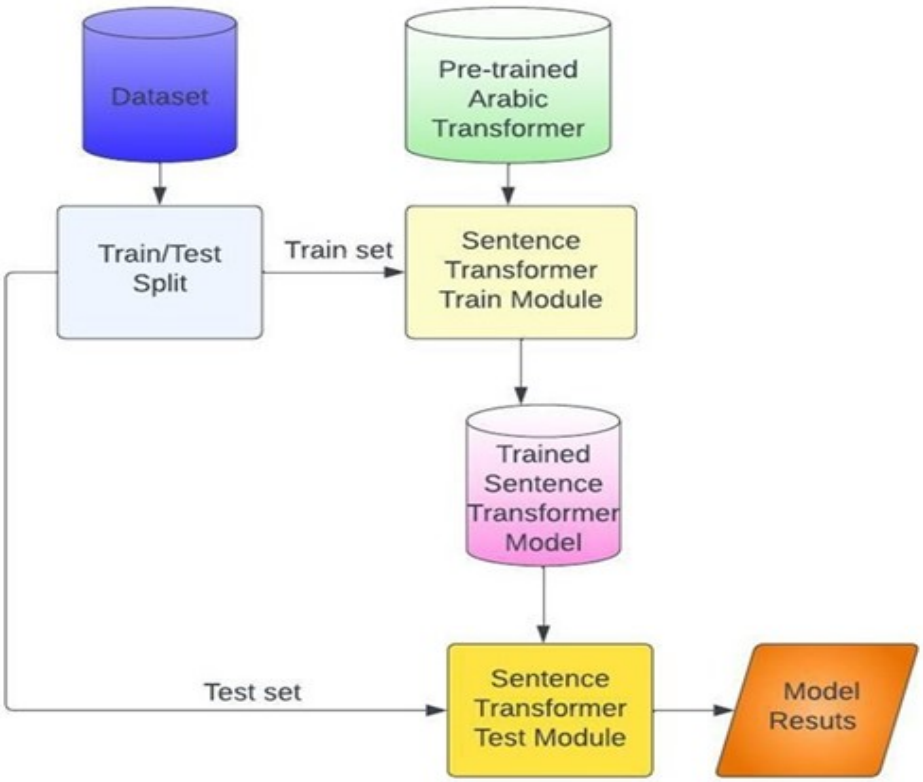


Fig. 2. The proposed model train/test flowchart.

standard train/test partition. Simultaneously, we integrate a pre-trained Arabic Transformer with the Sentence Transformer architecture. This integration ensures the model leverages the linguistic knowledge captured by the pre-trained transformer while optimizing for sentence-level embeddings. The Sentence Transformer is trained using the training portion of the ASAG dataset. This fine-tuning process adapts the model to capture the semantic nuances specific to Arabic short answers in academic contexts. After training, the resulting "ASAG Sentence Transformer Model" becomes a specialized tool for generating embeddings optimized for grading tasks. Performance evaluation uses the held-out test portion of the dataset, applying the trained model to generate embeddings and compute similarity measures. These embeddings capture rich semantic information, allowing for effective comparison between model answers and student responses.

Our approach employs four distinct similarity metrics to quantify the distance between embedding vectors:

$$d_{\text{euclidean}}(u,v) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2} \tag{6}$$

$$d_{\text{cosine}}(u,v) = \frac{u \cdot v}{||u||_2||v||_2} \tag{7}$$

$$d_{\text{manhattan}}(u,v) = \sum_{i=1}^{n}|u_i - v_i| \tag{8}$$

$$d_{\text{dot_product}}(u,v) = \sum_{i=1}^{n} u_i.v_i \tag{9}$$

V. Experiments and Results

A. Model Performance Comparison

The experimental results demonstrate that model architecture and size substantially impact performance across all datasets. As shown in Fig. 4, the 'aubmindlab-bert-large-arabertv02' model consistently achieves superior performance on DS2, with exceptional scores (cosine: 0.7985, manhattan: 0.7931) that represent the highest values in our evaluation. Similarly, Fig. 5 demonstrates this model's continued dominance on DS3 (cosine: 0.7955, manhattan: 0.7937). Interestingly, for DS1, Fig. 3 illustrates that the 'asafaya-bert-large-arabic' model outperforms all others, achieving the highest scores across all metrics (cosine: 0.7755, manhattan: 0.7701).

We observe a clear dataset difficulty gradient, with DS1 presenting more significant challenges for all models compared to DS2 and DS3. This pattern is evident when comparing Fig. 3, 4, and 5, where the overall height of the bars increases progressively from DS1 to DS3. This suggests inherent differences in linguistic patterns or semantic relationships across these datasets that affect model performance.

Regarding similarity metrics, cosine similarity consistently produces the most favorable results across all models and datasets, as visible in all three figures. This is followed closely by manhattan and euclidean distances. The dot product metric

(represented by orange bars in Fig. 3 to 5) consistently yields the lowest performance scores, particularly struggling with the UBC-NLP-AraT5v2-base model on DS1 (0.5395), as evident in Fig. 3.

The correlation coefficient choice also impacts results, with Spearman generally yielding higher values than Pearson, particularly for DS3, where the best model reaches 0.7998 for manhattan distance using Spearman correlation. While our figures display Pearson correlation results, comparative analysis between correlation types reveals this important distinction. These findings provide valuable insights for selecting appropriate Arabic language models and evaluation metrics for specific applications, highlighting the critical importance of considering both model architecture and evaluation dataset characteristics when assessing Arabic NLP model performance.

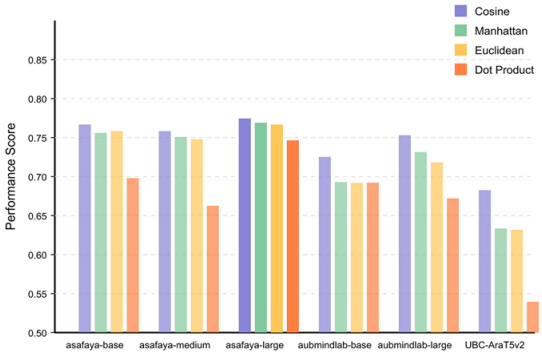


Fig. 3. DS1 performance scores for cosine, Manhattan, Euclidean and dot-product distance measurements.

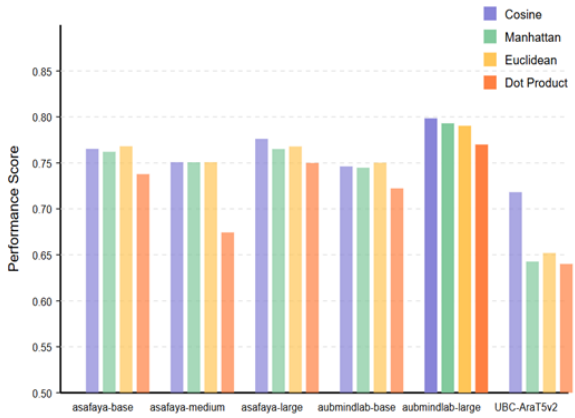


Fig. 4. DS2 performance scores for cosine, Manhattan, Euclidean and dot-product distance measurements.

B. Benchmarking with Previous Results

To assess the performance of our model in relation to established benchmarks, we conducted a comprehensive comparison using Pearson correlation coefficients. Table IV presents the comparative analysis between our best-performing model and previous benchmark results in the field of Automatic Short Answer Grading (ASAG).

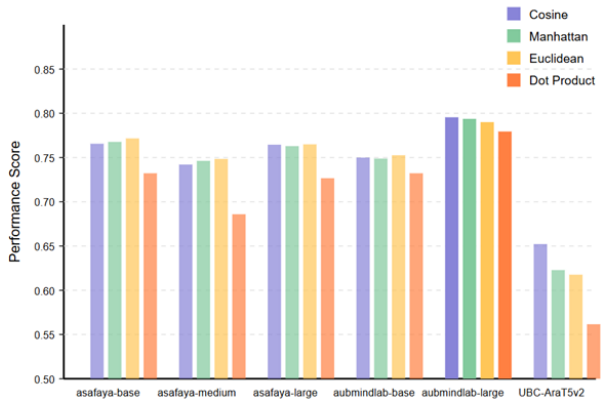


Fig. 5. DS3 performance scores for cosine, Manhattan, Euclidean and dot-product distance measurements.

Our proposed system demonstrates superior performance across all distance measures, with the arabertv02 + Cosine distance configuration achieving the highest Pearson coefficient of 0.7998. This outperforms all previous methodologies, including the AASAG models (Stemming model: 0.723, local weight: 0.731, and hybrid local global weight: 0.745) and exceeds the performance of Word2vec+Cosine + Light Stem (0.7758) from the Automatic Essay Scoring approach. The consistent high performance across different distance measures (Manhattan: 0.7937, Euclidean: 0.7904, and Dot-Product: 0.7794) further validates the robustness of our arabertv02-based approach.

The significant performance improvement (approximately 5.5% increase over the best previous method) can be attributed to several key factors:

1) Pre-training advantages: The arabertv02 model was pre-trained specifically on Arabic text, allowing it to capture nuanced semantic relationships particular to the language.

2) Contextual embeddings: Unlike static embeddings used in previous approaches, our model generates contextual embeddings that better capture the meaning of words based on their surrounding context.

3) Distance measure optimization: Our systematic evaluation of multiple distance measures allowed us to identify the optimal configuration for this specific task.

4) Large-scale transformer architecture: The arabertv02 architecture provides a more sophisticated representation of language compared to traditional methods, particularly beneficial for understanding complex answer structures.

These results highlight the effectiveness of our methodology and demonstrate its potential as a state-of-the-art solution for ASAG applications, particularly in Arabic language contexts.

VI. Discussion

Our proposed system demonstrates substantial improvements over existing Arabic grading approaches, achieving a 5.5% increase in correlation coefficients compared to previous best methods. This improvement stems from several key

TABLE IV. Comparison and Evaluation of Study Results Against Previous Models

Approach	Methodology	Pearson	Year
Proposed System	arabertv02 + Cosine	**0.7998**	2025
	arabertv02 + Manhattan	0.7937	
	arabertv02 + Euclidean	0.7904	
	arabertv02 + Dot-Product	0.7794	
AASAG	Stemming model	0.723	2023 [11]
	Local weight	0.731	
	Hybrid local & global	0.745	
Automatic Essay Scoring	WordNet+Cosine + Base	0.7469	2023 [6]
	WordNet+Cosine + Light	0.7553	
	Word2vec+Cosine + Base	0.7693	
	Word2vec+Cosine + Light	0.7758	
	BERT+Cosine + Base	0.7536	
	BERT+Cosine + Light	0.7616	
ASAG-Combined	Root Stem	0.7010	2020 [2]
	Light Stem	0.7037	
ASAG W-SM System	Root Stem	0.6830	2020 [2]
	Light Stem	0.6818	
ASAG-Basic System	Root Stem	0.6550	2020 [2]
	Light Stem	0.6340	

factors: AraBERT v02's pre-training on extensive Arabic corpora enables capture of nuanced semantic relationships specific to Arabic language patterns, providing significant advantages over generic multilingual models or traditional static embeddings.Unlike static representations used in previous approaches, our transformer-based system generates contextual embeddings that adapt meaning based on surrounding context, crucial for educational assessment where context determines answer correctness. Systematic evaluation of multiple distance measures reveals that cosine similarity provides optimal performance for Arabic semantic assessment, outperforming traditional Euclidean distance approaches commonly used in earlier systems. Progressive dataset enhancement through SemEval integration and reference answer inclusion provides more robust training signals, enabling better generalization across diverse question types and response patterns.

VII. Conclusion and Future Work

This study represents a significant advancement in the field of Automatic Short Answer Grading (ASAG) for Arabic language assessment. By leveraging the arabertv02 pre-trained language model with various distance measures, we have developed a system that substantially outperforms previous approaches. The experimental results demonstrate that our proposed methodology achieves state-of-the-art performance with a Pearson correlation coefficient of 0.7998 using the Cosine distance measure, representing a meaningful improvement over existing solutions. Our comprehensive evaluation has shown that transformer-based architectures specifically trained on Arabic text provide superior semantic understanding compared to traditional NLP approaches. The consistent performance across different distance metrics further validates the robustness of our approach. The integration of contextual word embeddings from arabertv02 has proven particularly effective at capturing the nuanced semantic relationships in Arabic short answers, enabling more accurate assessment that better aligns with human grading. This research contributes to the growing body of evidence supporting the efficacy of transformer-based models in educational technology applications, particularly for languages beyond English. Our findings have important implications for the development of automated assessment tools that can reduce the workload of educators while maintaining high standards of evaluation accuracy.

Future work will focus on expanding the evaluation to additional domains beyond cybersecurity, investigating cross-domain generalization capabilities, and exploring the integration of automated feedback generation systems. Additionally, we plan to conduct longitudinal studies on real classroom deployments to assess the practical impact of our approach in educational settings.

Data Availability

The AR-ASAG dataset used in this paper is available at: `https://data.mendeley.com/datasets/dj95jh332j/1`

References

[1] Reimers, Nils and Gurevych, Iryna, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *arXiv preprint arXiv:1908.10084*, 2019.

[2] Ouahrani, Leila and Bennouar, Djamal, "AR-ASAG an Arabic dataset for automatic short answer grading evaluation," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020.

[3] El Moatez Billah Nagoudi and Ferrero, Jérémy and Schwab, D. S., "LIM-LIG at SemEval-2017 Task1: Enhancing the Semantic Similarity for Arabic Sentences with Vectors Weighting," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 134–138, 2017, Association for Computational Linguistics.

[4] Wu, H. and Huang, H. and Jian, P. and Guo, Y. and Su, C., "BIT at SemEval-2017 Task 1: Using Semantic Information Space to Evaluate Semantic Textual Similarity," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, vol. L, pp. 77–84, 2017, Association for Computational Linguistics.

[5] Rohde, Douglas L. T. and Gonnerman, Laura M. and Plaut, David C., "An Improved Method for Deriving Word Meaning from Lexical," *Cognitive Psychology*, vol. 7, pp. 573–605, 2004.

[6] Meccawy, Maram and others, "Automatic Essay Scoring for Arabic Short Answer Questions using Text Mining Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, 2023.

[7] Rababah, H. and Al-Taani, A. T., "An automated scoring approach for Arabic short answers essay questions," in *Proceedings of 8th International Conference on Information Technology (ICIT)*, pp. 697–702, 2017, Amman, Jordan, doi: 10.1109/ICITECH.2017.8079930.

[8] Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina, "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*, pp. 4171–4186, 2019.

[9] Mikolov, Tomas and Chen, Kai and Corrado, Greg and Dean, Jeffrey, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.

[10] Shehab, A. and Faroun, M. and Rashad, M., "An Automatic Arabic Essay Grading System based on Text Similarity Algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 263–268, 2018, doi: 10.14569/IJACSA.2018.090337.

[11] Badry, Rasha M. and others, "Automatic Arabic Grading System for Short Answer Questions," *IEEE Access*, 2023.

[12] Abbas, Ayad R. and Al-qazaz, Ahmed S., "Automated Arabic essay scoring (aaes) using vectors space model (VSM) and latent semantics indexing (LSI)," *Engineering and Technology Journal*, vol. 33, no. 3, pp. 410–426, 2015.

[13] ElNaka, Abdelrahman and others, "AraScore: Investigating Response-Based Arabic Short Answer Scoring," *Procedia Computer Science*, vol. 189, pp. 282–291, 2021.

[14] Abdul Salam, Mustafa and Abd El-Fatah, Mohamed and Hassan, Naglaa Fathy, "Automatic grading for Arabic short answer questions using optimized deep learning model," *Plos One*, vol. 17, no. 8, pp. e0272269, 2022.

[15] Hochreiter, Sepp and Schmidhuber, Jürgen, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] Mirjalili, Seyedali, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[17] Abdeljaber, Hikmat A., "Automatic Arabic short answers scoring using longest common subsequence and Arabic WordNet," *IEEE Access*, vol. 9, pp. 76433–76445, 2021.

[18] Süzen, Neslihan and others, "Automatic short answer grading and feedback using text mining methods," *Procedia Computer Science*, vol. 169, pp. 726–743, 2020.

[19] Salton, G. and Wong, A. and Yang, C.-S., "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[20] Robertson, S. E. and Jones, K. Sparck, "Relevance weighting of search terms," *Journal of the American Society for Information Science*, vol. 27, no. 3, pp. 129–146, 1976.

[21] Baeza-Yates, Ricardo and Ribeiro-Neto, Berthier, *Modern Information Retrieval*, ACM Press, New York, NY, USA, 1999.

[22] Deerwester, Scott and Dumais, Susan T. and Furnas, George W. and Landauer, Thomas K. and Harshman, Richard, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[23] Elkateb, Sabri and Black, William and Rodríguez, Horacio and Alkhalifa, Musa and Vossen, Piek and Pease, Adam and Fellbaum, Christiane, "Building a wordnet for Arabic," in *Proceedings of The fifth international conference on Language Resources and Evaluation*, 2006.

[24] Hunt, James W. and McIlroy, M. Douglas, "An algorithm for differential file comparison," *Computing Science Technical Report*, Bell Laboratories, 1976.

[25] Cer, Daniel and others, "Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation," *arXiv preprint arXiv:1708.00055*, 2017.

A FOREX Trading System Based on Semi-Supervised News Classification, Market Sentiment Analysis, and GRU-CNN Deep Learning Models

Nabil MABROUK, Marouane CHIHAB, Younes CHIHAB
Computer Sciences Laboratory, Ibn Toufail University, Kenitra, Morocco

***Abstract*—Investors access the foreign exchange market (FOREX) not only to preserve their wealth but also to generate profits and achieve specific financial goals. It is one of the largest financial markets that investors rely on, and it is based on fluctuations in currency exchange rates to make a profit in different time cycles: short-term, medium-term, and long-term. In this article, we propose an automated FOREX trading system that combines two artificial intelligence algorithms: the first to classify news by pertinence (semi-supervised) and then to analyze market sentiment. This algorithm plays a crucial role in replacing the traditional fundamental analysis, which is based on macroeconomic factors, political events, and news headlines. The use of GAN-BERT helped improve performance in classification tasks with limited labeled data and reduced execution time. This algorithm demonstrates impressive results, achieving a high accuracy of 97.5%, which makes its output data more reliable for use in the second algorithm, a combination of two deep learning models: the Gated Recurrent Unit (GRU) and the Convolutional Neural Network (CNN). We enrich the dataset used in this phase with additional technical indicators and features that may help explain market fluctuations. We evaluated our final algorithm over multiple time frames and several windows; the results were impressive and confirmed by back-testing its potential profitability and risk.**

Keywords*—FOREX; trading; semi-supervised classification; sentiment analysis; machine learning; deep learning; RNN; CNN; GRU

I. Introduction

The Forex market, which is known as the largest financial market in the world [1], [2], [3], presents a profitable opportunity for investors with solid risk management and a good understanding of macroeconomic and political factors, as well as geopolitical events and technical analysis. Those factors cause the creation of price variations and exchange rates. However, to grow and preserve their wealth, investors take risks and participate in the FOREX market to take advantage of currency fluctuations. Due to geographical dispersion and 24-hour availability, investors can trade currencies at any time from different time zones around the world, resulting in high liquidity and a large volume of trades [4], [5]. Many national and international companies, individuals, and institutional investors utilize exchange rate volatility to achieve specific financial objectives [6]. However, to mitigate risks and avoid emotional investment, traders have developed a range of strategies based on various types of analysis.

A. Traditional FOREX Analysis

Since the early twentieth century, the world has witnessed the emergence of what we now refer to as traditional trading strategies [7]. They were deployed and adapted by traders to suit their trading strategy styles, as well as a few other tools that traders have found helpful in previous trading. Over the years, traditional Machine Learning (ML) methods have shown a strong ability to predict the trend of stock prices [8], [9], [10]. FOREX investors can be divided into two categories. The first relies on analyzing historical data to calculate indicators that can help explain price variation, then using them to predict price movement; but unfortunately, many factors influence the FOREX market and make it highly volatile, such as political events, economic data, world news, and market sentiments [11].

1) Technical analysis: Technical analysis is generally based on various techniques, including the use of points and graphs, such as candlestick graphs. It generated significant excess returns, at least until the early 1990s [12], [13], [14], [15]. According to [16] by Rico Nur et al., this type of analysis utilizes historical data to generate new technical indicators based on mathematical calculations. These indicators can reduce the noise created by high fluctuations and can also help detect patterns in past prices. These patterns form the foundation for creating trading rules that are important for generating trading signals that are (buy, sell, and hold) [17]. In fact, technical analysis is widely used, particularly in the foreign exchange market [18]. However, nowadays there exists an exhaustive list of technical indicators, but traders do not use them all at once; they use a combination of them that go with their trading style, and there is evidence that, at least until the early 1990s, it generated significant excess returns.

2) Fundamental analysis: Another way to forecast financial markets is to analyze the causes of those fluctuations. Expert investors examine macroeconomic and political factors, geopolitical events, financial news, market sentiments, and other factors to predict stock prices or trends [11], [19]. Generally, fundamental analysis is used in long-term predictions; investors choose currencies and stocks with strong economic indicators and favorable political climates for investment.

3) Sentiment analysis: The primary drivers of currency market fluctuations are news, economic indicators, and market sentiment [20]. Analyzing the general attitude of traders toward a particular FOREX market helps to understand the market's

behavior, making informed decisions, and minimizing the risk of loss. Optimistic emotions and expectations drive FOREX prices up, but pessimistic sentiment and news releases amplify the propensity to sell. Sentiment analysis increases the prediction accuracy of artificial intelligence algorithms [21].

In this study, we propose a comprehensive solution that leverages the three types of analysis, built on Algorithmic Trading techniques. Our system can automatically forecast and identify future price directions, trends, and cycles that characterize a FOREX dataset (time series dataset). In addition, it performs the optimal trade decision. Traditional trading strategies, data mining, and deep learning algorithms primarily rely on historical data that track price movements on the FOREX (OHLCV). These data are also enriched by adding more features based on technical indicators and trading rules for automatic or semi-automatic pattern recognition. In addition to that and to compensate for the fundamental analysis part that requires expert intervention, our proposed system in the first phase utilizes news articles and sentiment analysis to improve prediction accuracy. It utilizes a semi-supervised algorithm to resolve the issue of high-volume unlabeled data and also reduces time consumption. However, the primary goal of it is to classify and select the pertinent articles.

II. Related Works

In the past, the focus of many researchers has been concentrated on forecasting currency exchange rates, employing a variety of techniques and methodologies within this field. However, successful trading strategies have increasingly drawn attention to fundamental analysis. In this article, we will explore AI techniques, specifically deep learning and machine learning algorithms, while considering the three types of analysis. The first is technical analysis, grounded in mathematical calculations, whereas the second is fundamental analysis, which pertains to economic and political news, as well as other influencing factors [22]. The third is sentiment analysis, which can help improve forecasting accuracy [23].

Currency trading systems have been developed, allowing traders to execute all trading operations electronically through trading platforms [11]. This fact led to the availability of a vast volume of historical data, sparking an interest among researchers in resolving financial market problems using computational intelligence [24].

In 1967, Poole and, for the first time, utilized technical indicators to forecast Currency fluctuations [25], [26]. According to Poole and Dooley's viewpoint, currency exchange trading experiences are repeated over time, and historical data of past price trends can help shape them. They proved that, based on specific rules, traders could recognize past patterns and make profits; these rules enable traders to make informed decisions about when to buy or sell.

In addition to traditional trading strategies, another type of analysis has emerged, thanks to computer science, particularly artificial intelligence, and the high volume of historical data [27], resulting in numerous studies in this domain. The utilization of Data mining algorithms is for automatic or semi-automatic pattern recognition, establishing the connection and impact of different data [28], [29]. Machine learning (ML) algorithms have demonstrated significant efficacy in solving real-world categorization problems across multiple domains [30]. A variety of research teams have leveraged machine learning techniques to formulate trading strategies (Logistic regression, Random Forest, support vector machines, Neural Networks, etc.) [31].

In [22], using a linear regression model, they implemented a Forex forecasting system based on a sliding window of ten to 240 days of information on the seven major currency pairs; the model gets better results with the USD/CHF, USD/CAD, and EUR/USD, it was able to identify momentum opportunities to buy or sell and to decide if it is indifferent to buy or sell a currency. However, when it comes to classification problems, one of the classical machine learning methods is the Support Vector Machine (SVM), which still shows impressive results in solving significant classification problems, despite being mathematically complex and computationally expensive [32]. Many researchers have demonstrated the importance of this model in Forex trading. [33] Experimented with the advantages of the SVM model in forecasting Forex trends as a binary classification problem (uptrend or downtrend); The results showed that SVM models might help traders make automatic transaction decisions regarding Bid/Ask in the Forex Market. Additionally, Random Forest (RF) is widely used to resolve classification problems, especially in decision-making; it was introduced by Breiman in [34], who utilized bagging to enhance classification accuracy. In [35], the random forest algorithm is described as the top-performing algorithm for accurately predicting stock price directions. A small number of studies have been conducted on forecasting currency exchange rates using a random forest algorithm; one of these studies was carried out by [36]. In this article, the authors adopted the random forest algorithm to develop a trading strategy, and the proposed algorithm outperforms traditional technical indicators.

In recent years, the computing techniques of artificial intelligence, specifically Deep Neural Networks, have achieved significant advancements in natural language processing, image classification, voice translation, and other areas [37]. It is essential to highlight that specific Deep Neural Networks algorithms have been utilized for forecasting time series data and quantitative trading [38], [39], [40]. Deep learning algorithms have demonstrated improved performance and accuracy, yielding better returns in the finance sector [41], [42]. The broadening application of artificial intelligence has led to an increasing number of investors using deep learning models to predict and study stock and Forex prices. In [43], it has been demonstrated that fluctuations in stock and Forex prices can be predicted. Unlike traditional statistical and econometric models, deep learning can effectively describe complex influencing factors. The Forex, like any financial market, can be impacted by investor sentiment, especially fear, which leads to increased volatility or a shift towards safe-haven currencies, among other effects. However, numerous studies in the field of natural language processing (NLP) have studied sentiment analysis of textual content as a key to identifying public opinion and forecasting market trends. The majority of sentiment analysis techniques assess sentiment by calculating the polarity of words (positive, negative, neutral) or emotional words (such as happy, sad, like, dislike, etc.) that occur in textual documents [44]. In [45], they found an association between tweet sentiment and stock returns. They

utilize a simple machine learning algorithm (Naive Bayesian) to transform tweets into signals indicating whether to buy, hold, or sell. In [46], they correlate Facebook's daily sentiment with trading behavior. They exploit Facebook's Gross National Happiness Index (FGNHI), which measures people's mood by examining the positive and negative terms used by Facebook participants. In [46], they employed Facebook's daily sentiment index to prove the existence of a correlation between trading behavior and Facebook participants' mood. They exploit Facebook's Gross National Happiness Index (GNHI), which relates to positive and negative terms used. In [47], they utilized sentiment analysis to extract opinions from user-generated reviews, merging sentiment with content-based recommendation techniques to provide personalized product recommendations. They demonstrate the advantages of their method in various categories of Amazon products.

Recently, text classification algorithms have received increased attention in the literature thanks to the growing number of available resources. Text classification is the process of automatically organizing a set of documents into one or more predetermined or undetermined categories based on their topics [48]. On the same subject, the primary objective of supervised learning is to develop methods for classifying natural language processing documents [49]. The general challenge of text categorization can be subdivided into various sub-challenges, such as sentiment classification, functional classification, subject classification, and other types of classification [50]. In [51], an algorithm is created by integrating natural language processing, statistical pattern identification, and sentiment analysis to develop a system that forecasts the directional trend of a specific currency pair in the FOREX market based on the text of breaking financial news headlines. Their system has shown impressive results with a high accuracy level. Another research project was conducted by [52] to forecast foreign exchange trends based on news articles, specifically financial news, which are rich in essential factors widely used in fundamental analysis. However, they developed their algorithms utilizing the FinBERT language model, which is a specifically pre-trained NLP model geared towards the financial sector, for this initiative. FinBERT is developed by enhancing the BERT language model. Google Researchers were the initial publishers of BERT in [53], which is based on Transformer networks [54]. Its purpose is to pre-train deep bidirectional models from unlabeled text by simultaneously considering both left and right contexts across all layers [53]. BERT impressively improved the state-of-the-art for large language models.

When dealing with text classification problems, two traditional approaches are adopted: supervised and unsupervised learning. Recently, researchers have combined these two approaches into one named semi-supervised learning, which is a machine learning algorithm concerned with using both labeled and unlabeled data to perform specific learning tasks [55]; This method is beneficial when there is a large amount of unlabelled data accessible, but collecting labeled data is costly or time-consuming. It requires less human effort and provides greater precision [56].

III. Our Proposed Solution

Our proposed solution consists of two major phases. The first phase is a news classification and sentiment analysis system based on semi-supervised algorithms. The second phase involves a deep learning system that takes as input the results of the first algorithm, along with other features, to forecast trades and predict future price directions.

A. Data Description

As our approach is a sequence of two subsystems, we have built our strategy utilizing two datasets:

1) Dataset of the first phase: We have collected the dataset from various credible sources, including BBC, CNN, Yahoo Finance, CNBC, DailyForex, and FxStreet. These sources are specifically designed for financial markets and advertise millisecond-level delivery, helping minimize news publication delays at the source. Each row describes an article, identified by date and time, title, content, source, author, category, and subcategory (if labeled). The data cover the period from January 01, 2006, until May 30, 2025. This data set will be the input for a semi-supervised algorithm, which combines a small amount of labeled data, often expensive and time-consuming, with a large amount of unlabeled data.

a) Time-zone handling: The FOREX market operates 24 hours a day, 5 days a week. However, trading activity in the market is not constant throughout the entire day. Instead, trading activity is divided into several distinct sessions, each with its own characteristics and level of activity [57]. The Forex market is divided into three major regions: Australia-Asia, Europe, and North America [58]. During news data collection, to prevent information leakage and avoid inadvertently using future data to train the model, we utilized Coordinated Universal Time (UTC) as a normalized timestamp.

The objective of this phase is to classify news articles. At this stage, chronological order is not essential. To carefully maintain the distribution of topics in each split, we prepared a balanced dataset using the same-sized set of each class. For training, we used 100 examples of labeled data and 900 unlabeled examples of each class. To evaluate the performance of our model, we used 200 examples of labeled data of each class; 100 examples for validation and 100 examples for testing. Resulting in a total of 34,800 examples for the general categories.

2) Dataset of the second phase: We collected data on the two most traded currencies in the world: the US Dollar (USD) and the Euro (EUR). The EUR/USD currency pair represents the exchange rate between the euro (the base currency) and the US dollar (the quote currency), where the euro is used to purchase the US dollar. This dataset is structured as a time series, obtained through repeated measurements over time, such as hourly, daily, or weekly data. Each row of the dataset contains observations for five variables: Open, High, Low, Close, and Volume (OHLCV). The variables in the data set (OHLCV) are continuous. To be clear, this dataset is noisy due to high-frequency trading and redundant information; in a typical scenario, feeding these data as is to an LSTM or GRU model may lead to an overfitting problem during the training phase; and the model will predict the same last value

or a value very close to it; it may just memorize the previous data without learning. In our work, we tried to eliminate the noisy part of the dataset by first forecasting the next trend direction (classification problem) instead of the next price value (regression problem); the second step is training our model in different time cycles (short, medium, and long-term); the third is Minimizing the noisy parts of time series datasets; it can be achieved by utilizing a sliding window approach, which incorporates previous time steps as input. For example, a window size of four, which means that the model uses the current time step along with the three preceding time steps as input; see Fig. 1 for an illustrated example. In our article, we explore various window lengths, ranging from 1 to 15, to identify the most effective window size for our model. The fourth step involves generating additional features, specifically technical indicators, that are widely used to understand and analyze price movements in a specific market. However, technical indicators give an idea of where the price might go next in a given market at a specific time. We have used a large set of technical indicators generated over multiple periods with different parameters:

- The Weighted, Exponential, Simple, and Convergence Divergence Moving Average (WMA, EMA, SMA, and MACD).
- The Relatively Strength Index (RSI).
- The average directional index (ADX).
- The Commodity Channel Index (CCI).
- The Rate-of-Change (ROC).
- The Bollinger Band (BB), Moving Polynomial trending (MPT).

The goal of adding technical indicators is to improve the accuracy of our model; however, using a large set of features may lead to the opposite result. To avoid this, we reduced the high dimensionality of the dataset by retaining only the features that provide the most information and contribute the most to the prediction. According to Guyon [59], the objectives of feature selection are to reduce prediction time consumption, facilitate the understanding of predictors, and improve model accuracy. In [60], the authors have classified the majority of current feature selection algorithms into four primary categories: similarity-based methods, information-theoretic methods, sparse learning methods, and statistical methods.

The header of our final dataset is: DateTime, OHLCV variables, a set of selected technical indicators, sentiment_label, article_sentiment_score, article_source, Article_author.

3) Data splitting and prevention of information leakage: To prevent information leakage and ensure a realistic evaluation, we employed a strict chronological split for our data. The training set consisted of data from January 1, 2006, to December 31, 2019 (approximately 70% of the data). The validation set spanned January 1, 2020, to December 31, 2021 (approximately 10% of the data), and the test set covered January 1, 2022, to May 30, 2025 (approximately 20% of the data). Crucially, no data from the validation or test sets was used during training or feature engineering. All preprocessing steps were performed solely on the training data, and then the resulting transformations were applied to the validation and test sets. By maintaining a strict chronological order, we were able to more accurately assess the model's ability to generalize to unseen future data.

B. Phase 1: Semi-Supervised News Classification and Sentiment Analysis

Fundamental analysis is critical for FOREX traders. However, expert traders evaluate macroeconomic and political factors, fiscal policies, and other relevant factors to gain insight into future currency movements and develop strategies to capitalize on market trends. The high sensitivity to financial news makes the Forex market a risky environment for investors due to its high volatility and leverage, which can create fear in investors. This fear is an emotional driver that can amplify market volatility. In this phase, we have created two algorithms. The first algorithm utilizes news headlines to classify them based on their relevance, retaining only those that have an impact on FOREX trends. The second algorithm analyzes these news headlines and generates sentiment-based features.

1) GAN-BERT model for news classification: Let C be a set of categories, where n is the total number of categories in C, as shown in Eq. (1).

$$C = \{c_1, c_2, c_3, ..., c_n\} \tag{1}$$

The Article set belonging to each category forms the complete dataset where a_{ci} is the Article set attributed to the category a_i in the dataset; see Eq. (2).

$$A = \{a_{c1}, a_{c2}, a_{c3}...a_{cn}\} \tag{2}$$

Given a new text, "a_u" of an unknown category "c", the proposed model assigns the text to the most likely category from C. In [61], two models, a BERT-based model and a semi-supervised GAN [62], were combined to create a new model named GAN-BERT. Fig. 2 describes the architecture of the GAN-BERT model. As illustrated in the architecture of the GAN-BERT model (Fig. 2), the discriminator D classifies samples, while the generator G generates fake examples F. The discriminator utilizes the vector representations produced by BERT for unlabeled U and labeled L input texts. After training is complete, G is removed from the model, allowing the remaining components to be used for inference. The GAN-BERT model typically demonstrates better performance in classification tasks when there is a limited amount of labeled data. Another reason for employing GAN-BERT for news classification is that it takes into account not only news writing styles, but also potential fake writing styles generated by the fake data produced by the generator.

2) FinBERT for news sentiment analysis: The specialized language and the scarcity of labeled data in the financial market domain hinder the effectiveness of general sentiment analysis models. The author in [63], proposed FinBERT, a BERT-based model designed to address NLP tasks specific to the financial sector. In our study, we fine-tune FinBERT using a small training set to create a new feature that enriches our datasets with sentiment labels for the FOREX market (positive, negative, and neutral). It is important to note that sentiments can be expressed by both experts and beginners. Therefore, we

index	feature	Target
t_6	X_6	Y_6
t_7	X_7	Y_7
t_8	X_8	Y_8
t_9	X_9	Y_9

index	feature	Target
t_6	X_3 X_4 X_5 X_6	Y_6
t_7	X_4 X_5 X_6 X_7	Y_7
t_8	X_5 X_6 X_7 X_8	Y_8
t_9	X_6 X_7 X_8 X_9	Y_9

Fig. 1. Example of sliding-window approach.

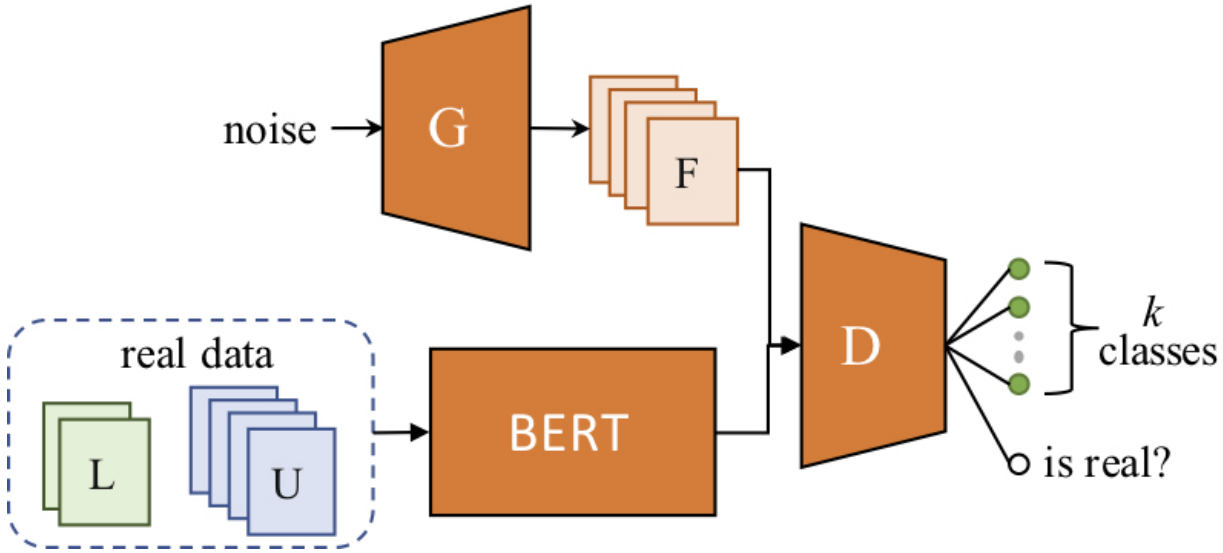


Fig. 2. GAN-BERT architecture.

will also consider the authors of the articles and assign a score to each based on their accuracy in making correctly expressed predictions.

3) GanBERT configuration and training hyperparameters:

- Base Transformer Model: BERT-base-cased.
- Discriminator Architecture. Consisted of the pre-trained BERT-base-cased model, fine-tuned for two tasks:
  - Binary Classification (Real vs. Fake): Distinguishing real news articles from generated articles.
  - Multi-Class Topic Classification: Classifying real news articles into one of the [Number of Topics] predefined topic categories.
- Output layer:
  - Real vs. Fake: Linear layer with sigmoid activation function.
  - Topic Classification: A separate linear layer with a Softmax activation function (for the multi-class classification task).
- Generator Architecture: A multi-layer perceptron (MLP) that takes random noise as input and generates synthetic feature vectors designed to resemble those of minority-class news articles. This helps to improve the discriminator's ability to classify minority topics, improving the accuracy of classification training.
- Layers: Two fully connected layers with 512 units, followed by a linear layer projected to the BERT embedding dimension (768).
- Activation Function: LeakyReLU activation (with a negative slope of 0.2) was used for the hidden layers of the generator.
- Training hyperparameters: Table I summarizes the key hyperparameters used in training the GanBERT

model for news classification. These parameters were carefully chosen to optimize the performance of the model and prevent overfitting.

TABLE I. GANBERT TRAINING HYPERPARAMETERS

Hyperparameter	Value
Learning Rate (Disc.)	7e-7
Learning Rate (Gen.)	5e-4
Optimizer	Adam
Batch Size	64
Latent Dimension	100
Discriminator Dropout	0.2
Generator Dropout	0.2
Epochs	50
Early Stopping Patience	6
Loss Function (Disc.)	Binary Cross-Entropy
Loss Function (Class.)	Categorical Cross-Entropy

4) FinBERT Fine-Tuning Hyperparameters:

- Learning Rate: 2e-5
- Batch Size: 32
- Number of Epochs: 3
- Weight Decay: 0.01
- Optimizer: AdamW with a weight decay of 0.01

C. Phase 2: Forecasting Future Price Direction (CNN-GRU APPROACH)

1) Target variable: To get rid of the FOREX market disturbances and reduce high price variation, we reformulated our problem from regression into classification; In this article, we aim to forecast the next trend direction or next price direction; the price in the original dataset is a continuous variable, we had to generate a new categorical variable named Target based on specific rules related to our strategy; this new variable indicates our final decision, which will be a buy, sell, or hold signal.

Each operation is associated with transaction costs, which can be more expensive than returns. For that issue, we used a threshold called α; When the log return of the current day is inferior to -α, it indicates that it is an uptrend, so it is a SELL signal, and when the log return is superior to α, it indicates that it is a downtrend, so it is a BUY signal; otherwise, it is a HOLD signal. We encoded this feature using one-hot encoding; see Eq. (3). Another application of the threshold α is to address the issue of unbalanced classes within datasets, a challenge that we encounter when α is set too low or too high. However, artificially balancing the dataset can negatively impact sequential datasets. To explore this, we ran our algorithm using various thresholds, starting from zero (which created a binary classification problem) and increasing up to 0.01. Otherwise, when we increased the threshold to 0.01, we again encountered the issue of unbalanced datasets.

Before model training, we addressed the challenge of class imbalance by optimizing the log-returns threshold (α). Through experimentation, we identified a threshold value of 0.0014, which yielded a dataset with approximately equal representation across the three classes: 1598 BUY, 1585 SELL, and 1593 HOLD signals. This nearly balanced distribution was deemed crucial to ensure robust and unbiased model performance.

$$Target = \begin{cases} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, & \text{if } Log(price_t/Price_{t-1}) > \alpha \quad "BUY" \\ \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}, & \text{if } Log(price_t/Price_{t-1}) < -\alpha \quad "SELL" \\ \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, & \text{Otherwise} \quad \text{"HOLD"}. \end{cases} \tag{3}$$

2) Recurrent neural networks LSTM and GRU: To deal with sequential data problems, such as time series or text, researchers have developed the Recurrent Neural Network (RNN) [64], [65], [66], which is a specialized type of neural network that possesses a form of memory that can retain information from previous input. Thanks to the feedback connections in their architecture, which facilitate the retention of information across time steps. RNNs are particularly adept for sequence modeling tasks, where the temporal context of data points is crucial for accurate prediction and understanding.

Long-Short-Term Memory (LSTM), published by [67], [68], is an architecture of recurrent neural networks (RNNs) that can help address the vanishing gradient problem faced by standard RNNs. LSTM can effectively bridge time lags exceeding 1,000 discrete time steps. This capability makes them suitable for tasks where long-range temporal dependencies are critical [67], [68], [69], [70].

A Gated Recurrent Unit (GRU) is a specific type of recurrent neural network unit within the broader category of deep neural networks. It was introduced by [71] to allow each recurrent unit to capture dependencies across multiple time scales adaptively. Recently, recurrent neural networks have shown promising results in various machine learning tasks, particularly when dealing with input and output of variable duration [72]. The GRU is recognized for its effectiveness in sequence learning tasks as it addresses the challenges of vanishing and exploding gradients that are often encountered in traditional recurrent neural networks (RNNs), making it well-suited for learning long-term dependencies.

3) Convolutional Neural Network: From a mathematical operation called convolution, the researchers derived the name of a new type of deep neural network, known as the convolutional neural network (CNN). It involves the linear interaction between matrices. The connectivity patterns of its neurons are modeled after the organization of the visual cortex in animals. This convolution technique enables the recognition of visual patterns in input data [73]. Like other artificial neural networks, CNNs consist of multiple layers, including the convolutional layer, which is the most critical. Each convolutional layer identifies features by sliding various filters over the input matrix and conducting piece-by-piece comparisons [74]. As indicated in [75], the two most significant advantages of convolutional neural networks are their ability to tackle complex tasks with a smaller number of parameters in the ANN and their ability to extract abstract features as data progress through deeper layers.

4) GRU-CNN model architecture: By combining the gated recurrent network and the convolutional neural network approaches, we developed our model. As shown in Fig. 3, it is composed of six layers:

- The input is a 3D tensor of shape (samples, timesteps, features). Each sample is a window of historical data. The time steps are the number of data points in a single window, and the features represent the values of all variables on the days to look back.
- The 1D convolutional layer: It takes the output of the previous layer as input; it automatically learns features by sliding multiple filters over the input matrix to extract local patterns.
    - Filters: 64
    - Kernel Size: 2
    - Activation: ReLU
- The third layer is the GRU layer; After the CNN, a GRU layer processes the output to capture the sequential nature of the patterns within the window. It is implemented in the many-to-one architecture, as shown in Fig. 4, because the goal is to predict the next price direction, not multiple prices. This layer is added to deal with the nature of sequential data.
    - Units: 64
    - Activation: tanh
    - Recurrent Activation: sigmoid
    - Dropout: 0.3
- The Dense Layer 1 (Fully Connected): The GRU output is flattened and fed into a dense layer.
    - Units: 32
    - Activation: ReLU
- The Dense Layer 2 (Fully Connected): The GRU output is flattened and fed into a dense layer.
    - Units: 8
    - Activation: ReLU
- The last layer is the output layer with three units; it provides a single decision (buy, sell, or hold).
    - Units: 3
    - Activation: Softmax

GRU-CNN Training Hyperparameters

- Window size: 1 to 10 days.
- Epochs: 50
- Batch Size: 64
- Learning Rate: 0.0005
- Optimizer: Adam
- Loss Function: Categorical Cross-Entropy
- Early Stopping Patience: 6

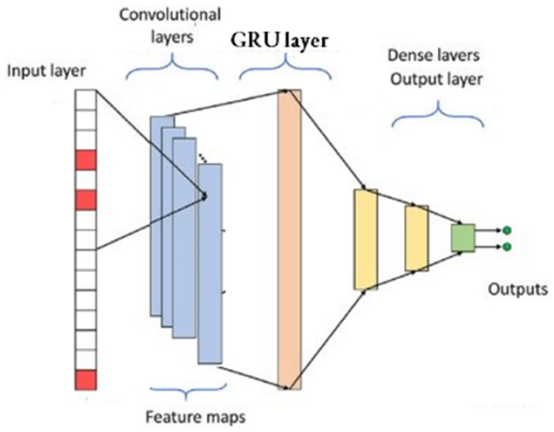


Fig. 3. GRU-CNN architecture.

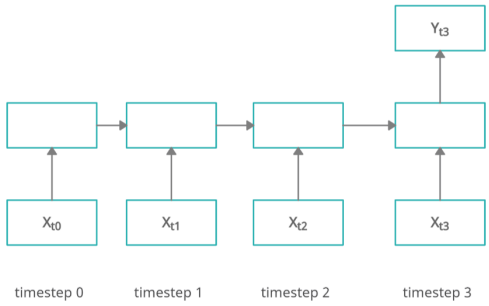


Fig. 4. Many to one GRU approach.

D. Phase 3: Our Investment Strategy and Decision Making

To develop an effective trading strategy, we must first determine the goal of our strategy: to generate short-term profits or build long-term wealth, and identify a personal risk profile or hedge against market volatility. Understanding whether we are more comfortable with conservative, moderate, or aggressive trading approaches will guide our decision-making process. Finally, we must evaluate the available time and resources.

In FOREX trading, the currency exchange rates are generally relatively low. This fact impacts the profitability of investments, particularly when the invested amounts are not significant; The returns may be less than the associated transaction costs. To resolve this issue, a third party, known as brokers in the FOREX market, offers a method called “leverage,” which involves borrowing funds to acquire an asset. In our strategy, we employed a leverage of 1:100, allowing us to invest 1,000$ while trading up to 100,000$. Our investment strategy is detailed in Fig. 9. Our aim is to buy when prices are low and sell when they increase, which means that our machine learning model needs to predict the future direction of trades.

To facilitate this, we developed an algorithm that compares the logarithmic return of the current day against a threshold, α. If the return exceeds α, we proceed with a buy; if it falls below $-\alpha$, we opt to sell; and if it lies between $-\alpha$ and α, we hold our position. In this context, the variable we are considering is categorical, so we applied one-hot encoding to prepare the data for our algorithms and fit our model effectively.

In addition to the threshold α, risk management rules are necessary to avoid stacking in the hold position; Stop-loss and take-profit are calculated based on the entry point, unlike the threshold α, which is compared to the log return of the current day.

IV. Discussion and Results

For the news classification phase, which serves as a fundamental analysis, it is the most important and basic phase in our strategy; all other parts of the system rely on it. We trained our model for 50 epochs, with the option of early stopping. As shown in Fig. 5 and 6, the model was well trained. The goal is to classify news articles into general classes, including sports, technology, business, health, politics, and finance. Subsequently, we conducted a secondary categorization to identify subcategories, including stock market, banking, personal finance, investment, and economic indicators. The GAN-BERT model achieved an impressive accuracy of 97.5%, demonstrating both efficiency and performance; see Fig. 6. As illustrated in Fig. 5, the training loss line indicates that the model fits the training data well and can be generalized to new data, as confirmed by the validation loss line.

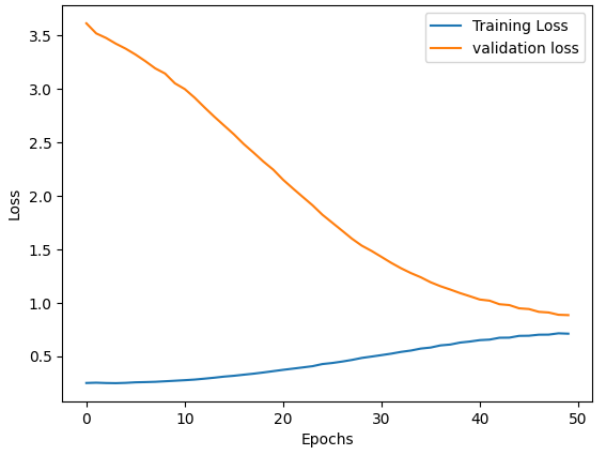


Fig. 5. GANBERT training validation loss.

The high efficiency of this model, as we have seen, makes the classified data a reliable source for the next stage. However, we have improved the persistence of these data by performing another NLP task, represented by fine-tuning the FinBERT model, which is specifically developed for the financial sector. We have executed it to label our dataset with the FOREX market sentiment, this variable is represented with three labels (Positive, Negative, and Neutral); by adding the sentiment feature, our dataset becomes able to describe the market mood and attitude of traders and investors towards the Forex market.

In the second phase, which focuses on forecasting future trends, we employed a combined CNN and RNN model,

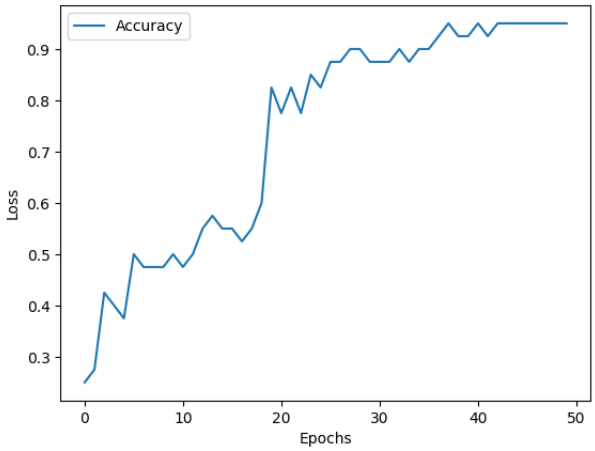


Fig. 6. GANBERT validation accuracy.

specifically a CNN and GRU architecture. We trained this model over 50 epochs, varying window lengths (1 day, 2 days, 3 days, 4 days, 5 days and 10 days), as shown in Fig. 10 to 19; by analyzing the training and validation loss graphics, we can see that our model fits the training data well and can be generalized for a larger time frame. Furthermore, the training validation accuracy graphics illustrated promising results in terms of both accuracy and precision.

We conducted the experiment of our model over several time frames (1 day, 5 days, 10 days, and 15 days), as outlined in Table II. In general, the model achieved higher accuracy in forecasting the future trend direction.

In forecasting the price direction for the next day, our model was less accurate when using a window length of one day, achieving only 58% accuracy. This is due to the time it takes for the event to impact the FOREX market and the number of news articles that we used. However, when we used a window length of two days, we observed a slight improvement in the performance of the model, achieving an accuracy of 80%. Although using a 3-day window length, the results are more promising, achieving 83% accuracy, and continue to improve until a 10-day window length, when it decreases from 85% to 81%.

In the second experiment, we attempted to forecast the price for the next week (5 days). Although using a 1-day window length, unlike in the first experiment, the model's performance showed better results, with an accuracy of 81%. The wider we expanded the window length, the more accurate the model became, achieving 93% accuracy using a 5-day window length, and it starts to decrease with a 10-day window length.

In the third experiment, we attempted to forecast the direction of the next 10 days' prices (i.e. over the next two weeks, excluding non-business days). We obtained results similar to those of the last experiment. It achieved an accuracy between 83% and 92%. To conclude, the model was more accurate in forecasting the next week (5 days) trend direction, especially when the window length was 5 days. Fine-tuning our model is necessary to improve both the accuracy and the running time of our model simultaneously. However, Fig. 11, 13, 15, 17

TABLE II. MODEL ACCURACY BY WINDOW-SIZE AND TIME FRAME

Target	Window lenght					
	1 day	2 days	3 days	4 days	5 days	10 days
1 day	0.5876	0.8050	0.8387	0.8422	0.8511	0.8185
5 days	0.8092	0.8098	0.8321	0.8450	0.9291	0.9156
10 days	0.8341	0.8383	0.8667	0.8943	0.9165	0.9023

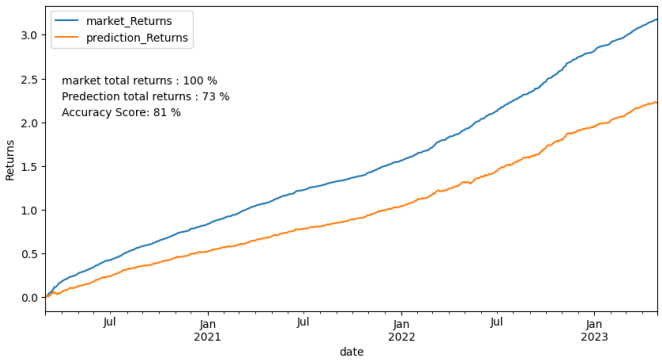


Fig. 7. The cumulative profits during the COVID-19 pandemic.

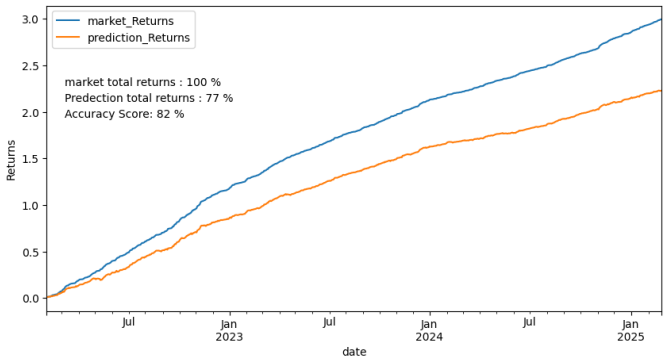


Fig. 8. The cumulative profits during the Russia-Ukraine war.

TABLE III. DIEBOLD-MARIANO TEST RESULTS (P-VALUES)

Model Pair	DM Test Statistic	p-value	Conclusion ($\alpha = 0.05$)
GRU-CNN vs Naïve Bayes	7.61	1.60×10^{-11}	Significant (GRU-CNN is better)
GRU-CNN vs Logistic Regression	9.84	2.44×10^{-16}	Significant (GRU-CNN is better)
GRU-CNN vs Random Forest	7.86	4.82×10^{-12}	Significant (GRU-CNN is better)
GRU-CNN vs XGBoost	3.19	1.91×10^{-03}	Significant (GRU-CNN is better)

and 19 show the training loss at each epoch; we notice that the training loss plot decreases to a point of stability.

To conduct a rigorous statistical analysis of our models' performance, we employed the Diebold-Mariano (DM) test to compare the predictive accuracy of our hybrid GRU-CNN model against the baseline models. The DM test evaluates the null hypothesis (H_0) that two models have the same forecast accuracy. A key component of the test is the zero-one loss function, which assigns a loss of 0 for a correct prediction and 1 for an incorrect one. A p-value below the chosen significance level (α=0.05) indicates that the difference in performance is statistically significant.

The results of the Diebold-Mariano test are summarized in Table III. The p-values for all comparisons are exceptionally low, all falling well below the 0.05 significance threshold. This finding provides strong statistical evidence that our GRU-CNN model is a better predictor of this task compared to all traditional machine learning benchmarks. The low p-values confirm that the performance gains of the hybrid model are not a result of random chance. Still, they are instead due to its inherent ability to capture more complex patterns in the financial time series data. This analysis validates our choice of a deep learning approach for this problem and confirms its effectiveness.

A. The Effectiveness of Our System During Times of Crisis

In times of crisis, as investors panic, the index of fear increases, causing adverse ripple effects throughout the economy, and the volatility of the market spikes dramatically. Each speculative attack on the Forex market often accompanies a crisis. Generally, we have tested our proposed solution over a long period that has witnessed several crises, and during this period, our system has consistently shown promising results. However, to confirm the effectiveness of our system during crisis times, we backtested our strategy on two crises: the COVID-19 pandemic from March 2020 to May 2023, and the Russia-Ukraine War, which began in February 2022 and is still ongoing. As shown in Fig. 7, which illustrates the cumulative profits during the COVID-19 pandemic, our system successfully predicted the future direction of the prices, achieving an accuracy of 81% with a total profit of 73% compared to the ideal. A similar result can be observed in Fig. 8, which illustrates cumulative profits during the Russia-Ukraine War; the accuracy was around 82% and the total profits were 77%; These results, while promising, should be interpreted with consideration for the complexities of real-world FOREX trading. Although our primary focus in this article is on demonstrating the effectiveness of integrating semi-supervised news classification, sentiment analysis, and time-series modeling in forecasting FOREX price directions, it is essential to acknowledge the potential impact of transaction costs (spreads), slippage, and inherent risks. To provide a more realistic perspective, we have applied a moderate estimate of a 10% reduction to the backtested profits achieved during both the COVID-19 pandemic and the Russia-Ukraine war. This reduction 10% accounts for the aggregate impact of transaction costs, primarily spreads on traded currency pairs, and

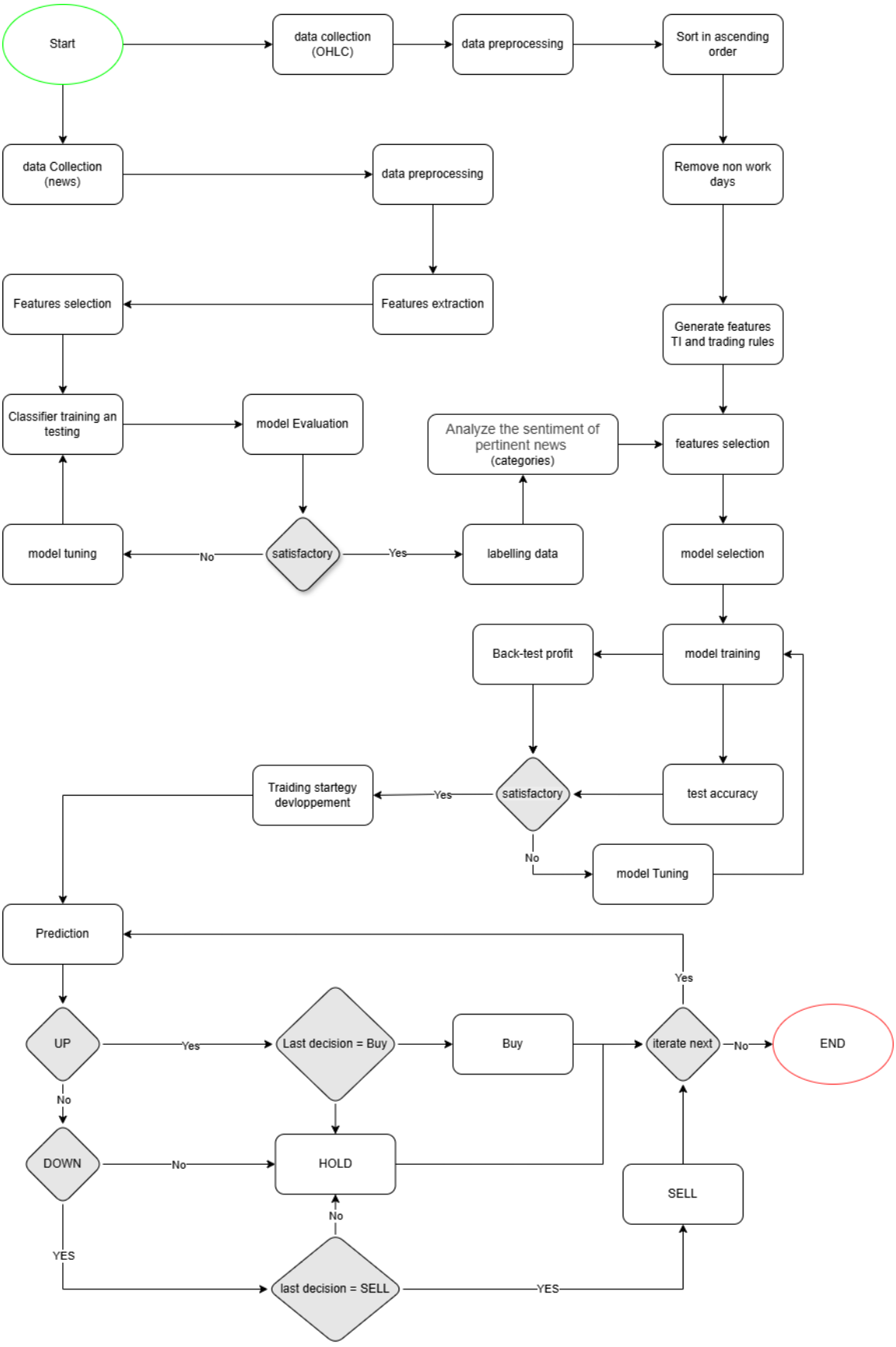


Fig. 9. Our investment strategy for trading foreign exchange.

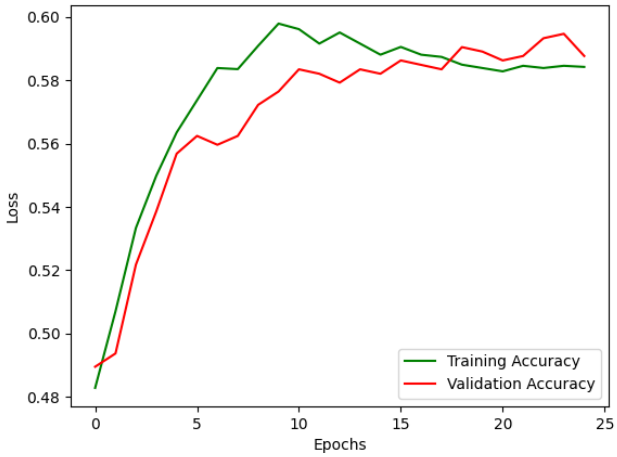


Fig. 10. CNN-GRU Training validation Accuracy 1 day window length.

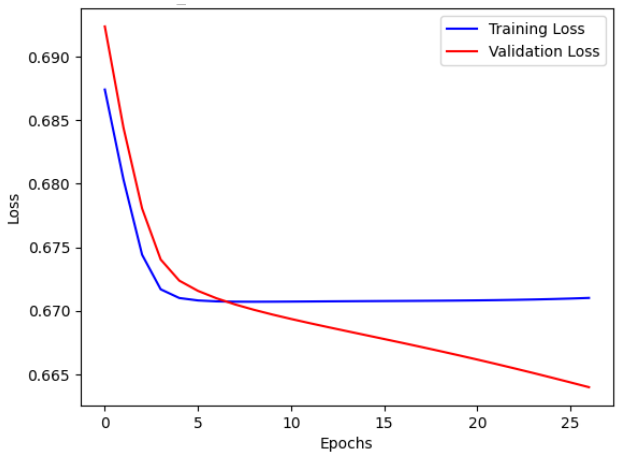


Fig. 11. CNN-GRU Training validation loss 1 day window length.

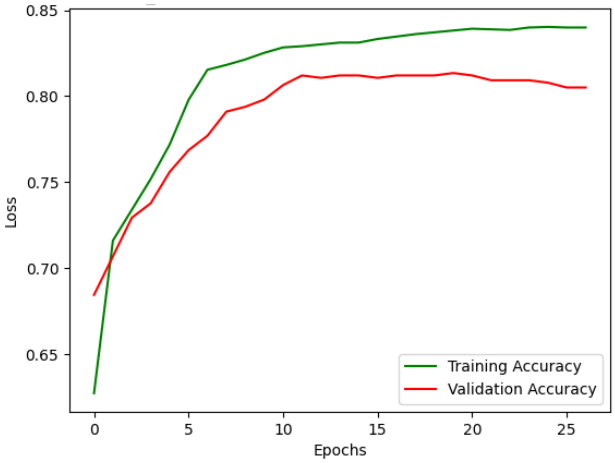


Fig. 12. CNN-GRU Training validation Accuracy 2 days window length.

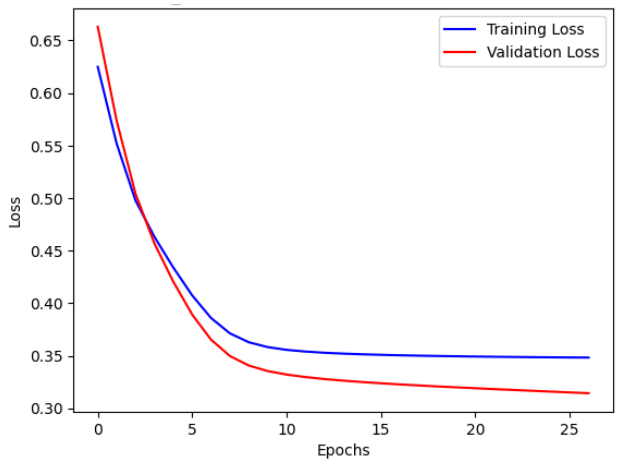


Fig. 13. CNN-GRU Training validation loss 2 days window length.

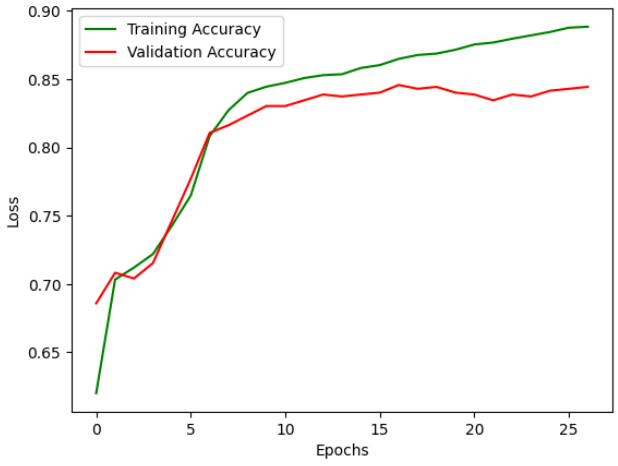


Fig. 14. CNN-GRU Training validation Accuracy 3 days window length.

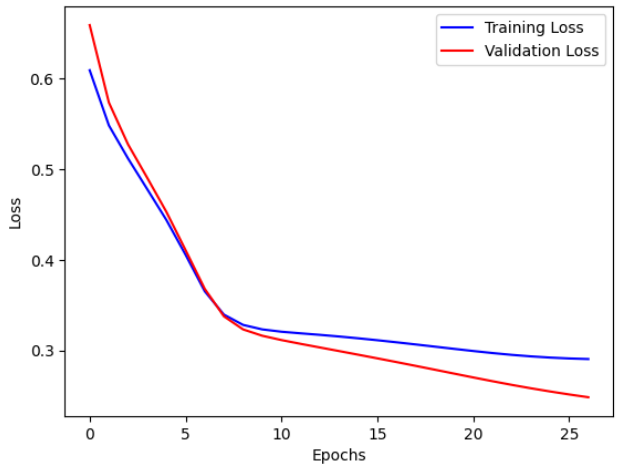


Fig. 15. CNN-GRU Training validation loss 3 days window length.

estimated slippage derived from historical volatility patterns. We maintain that even with this moderate adjustment, the system shows a strong potential for profitability and resilience in forecasting FOREX price directions during turbulent market conditions. By analyzing these two results, we can conclude that our proposed strategy can overcome obstacles during crisis

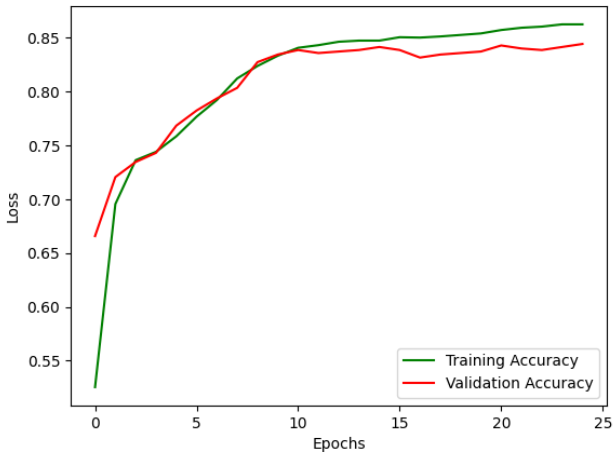


Fig. 16. CNN-GRU Train validation Accuracy 5 days window length.

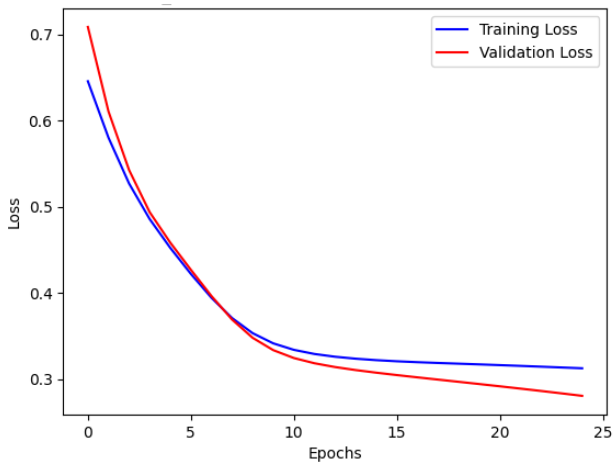


Fig. 17. CNN-GRU Training validation loss 5 days window length.

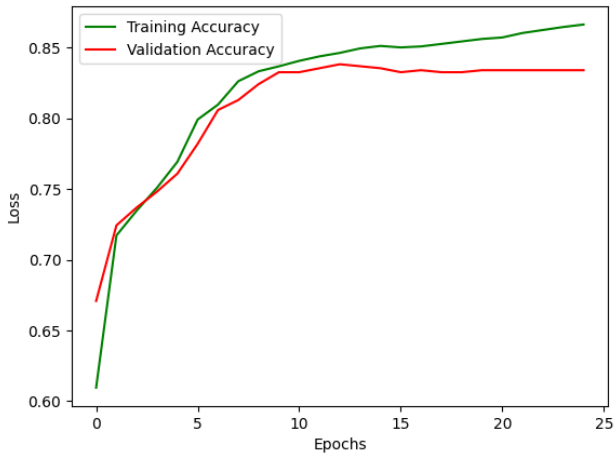


Fig. 18. CNN-GRU Training validation Accuracy 10 days window length.

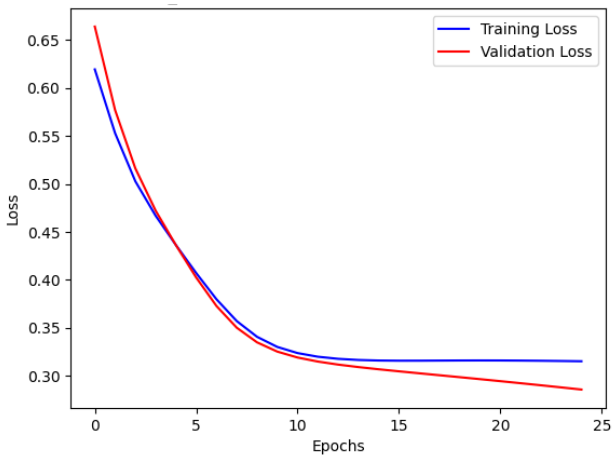


Fig. 19. CNN-GRU Training validation loss 10 days window length.

times by imitating fundamental analysis through the analysis of news headlines and social media posts. Future research will incorporate a more detailed, high-fidelity simulation incorporating granular transaction cost and slippage data, as well as a thorough risk assessment using industry-standard metrics, such as Sharpe and Sortino ratios, to validate these findings further and provide a more complete picture of the system's performance across diverse market environments.

V. Conclusion

In the FOREX market, numerous factors can affect the state of the market in various ways, making it too complex to develop an optimal trading strategy. This study presents a comprehensive system for automated Forex trading, utilizing an NLP-based strategy to analyze sentiment and forecast the expected impact on the Forex market through news reports. The results demonstrate that the system exhibits high profitability and can be effectively deployed for real-life trading purposes, overcoming external events such as regime changes, crisis times, epidemics, and natural disasters. This solution was developed and back-tested for a specific period; it may not remain helpful and profitable in the future.

Data Availability Statement

Due to privacy and ethical restrictions. The data that support the findings of this study cannot be made publicly available. However, a simplified version of the codebase is available on request from the corresponding author.

References

[1] T. Keeris, "the world's largest financial market: forex," *Social Science Research Network*, p. 2, 2018.

[2] T. Mahmud, T. Akter, S. Anwar, M. T. Aziz, M. S. Hossain, and K. Andersson, "Predictive modeling in forex trading: A time series analysis approach," in *2024 Second International Conference on Inventive Computing and Informatics (ICICI)*. 2024 Second International Conference on Inventive Computing and Informatics (ICICI), 2024, p. 390–397.

[3] A. Chaboud, D. Rime, and V. Sushko, *The foreign exchange market*. Edward Elgar Publishing, 2023.

[4] H. Law, S. Aghabozorgi, S. Lim, Y. Teh, and T. Herawan, "Processing of kuala lumpur stock exchange resident on hadoop mapreduce," in *Proceedings of the International Conference on Data Science (ICDATA)*. Proceedings of the International Conference on Data Science (ICDATA), 2014, p. 1.

[5] G. McLeod, "Forex market size: A traders advantage," *DailyFX (January 23).*, 2014.

[6] L. Puspitawati, "Information system for forex investment and their effects on investment growth in foreign currencies," *International Journal of Research and Applied Technology (INJURATECH)*, 2021.

[7] Y. Pei, J. Cartlidge, A. Mandal, D. Gold, E. Marcilio, and R. Mazzon, "Cross-modal temporal fusion for financial market forecasting," *arXiv preprint arXiv:2504.13522*, 2025.

[8] L. Malagrino, N. Roman, and A. Monteiro, "Forecasting stock market index daily direction: A bayesian network approach," *Expert Systems with Applications*, vol. 105, pp. 11–22, Sep. 2018.

[9] M. M. Kumbure, C. Lohrmann, P. Luukka, and J. Porras, "Machine learning techniques and data for stock market forecasting: A literature review," *Expert Systems with Applications*, vol. 197, p. 116659, 2022.

[10] P. K. Sarangi, M. Chawla, P. Ghosh, S. Singh, and P. K. Singh, "Forex trend analysis using machine learning techniques: Inr vs usd currency exchange rate using ann-ga hybrid approach," *Materials Today: Proceedings*, vol. 49, p. 3170–3176, 2022.

[11] L. Abednego, "Forex trading robot with technical and fundamental analysis," *Journal of Computers*, p. 1089–1097, 2018.

[12] C. Neely, J. Ulrich, and P. Weller, "The adaptive markets hypothesis: Evidence from the foreign exchange market,," *J. Financ. Quant. Anal*, vol. 44, p. 467–488, 2009.

[13] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A systematic review of fundamental and technical analysis of stock market predictions," *Artificial Intelligence Review*, vol. 53, p. 3007–3057, 2020.

[14] R. Pramudya and S. Ichsani, "Efficiency of technical analysis for the stock trading," *International Journal of Finance & Banking Studies*, vol. 9, p. 58–67, 2020.

[15] R. Schabacker, *Technical analysis and stock market profits*. Harriman House Limited, 2021.

[16] R. N. Ilham, I. Sinta, and M. Sinurat, "The effect of technical analysis on cryptocurrency investment returns with the 5 (five) highest market capitalizations in indonesia," *Jurnal Ekonomi*, vol. 11, p. 1022–1035, 2022.

[17] S. Galeshchuk and S. Mukherjee, *FOREX Trading Strategy Optimization*. Springer International Publishing, 2018.

[18] M. Fisichella and F. Garolla, "Can deep learning improve technical analysis of forex data to predict future price movements?" *Ieee Access*, vol. 9, p. 153083–153101, 2021.

[19] K. Cao and H. You, "Fundamental analysis via machine learning," *Financial Analysts Journal*, vol. 80, p. 74–98, 2024.

[20] L. Abednego, C. E. Nugraheni, and A. Fedora, "Forex sentiment analysis with python," *International Journal of Advanced Research in Economics and Finance*, vol. 4, p. 46–55, 2022.

[21] Y. Dave, "Predicting forex pair movements: Integrating sentiment analysis, technical, and fundamental indicators using machine learning and deep learning models," 2024.

[22] C. Maté, "Forecasting in forex the spot price interval of tomorrow with the same information of today. an analysis of the seven majors using a linear regression model based on interval arithmetic," *Knowledge-Based Systems*, vol. 258, p. 109923, Dec. 2022.

[23] J. Saettele, *Sentiment in the forex market: indicators and strategies to profit from crowd behavior and market extremes*. Wiley Online Library, 2021.

[24] E. Gerlein, M. Mcginnity, A. Belatreche, and S. Coleman, "Evaluating machine learning classification for financial trading: An empirical approach," *Expert Systems with Applications*, vol. 54, pp. 193–207, Jul. 2016.

[25] W. Poole, "Speculative prices as random walks: An analysis of ten time series of flexible exchange rates," *Southern Economic Journal*, vol. 33, pp. 468–478, 1967.

[26] R. Hudson and A. Urquhart, "Technical trading and cryptocurrencies," *Annals of Operations Research*, vol. 297, p. 191–220, 2021.

[27] R. Moscinski and D. Zakrzewska, *Building an Efficient Evolutionary Algorithm for Forex Market Predictions*. Poland: Springer, 2015.

[28] I. Witten, E. Frank, and M. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier, 2011.

[29] W. Haoxiang, S. Smys *et al.*, "Big data analysis and perturbation using data mining algorithm," *Journal of Soft Computing Paradigm (JSCP)*, vol. 3, p. 19–28, 2021.

[30] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN computer science*, vol. 2, p. 160, 2021.

[31] N. MABROUK, M. CHIHAB, Z. HACHKAR, and Y. CHIHAB, "Intraday trading strategy based on gated recurrent unit and convolutional neural network: forecasting daily price direction," *International Journal of Advanced Computer Science and Applications*, vol. 13, 2022.

[32] S. Suthaharan, *Support Vector Machine*. Boston, MA: Springer US, 2016, vol. 36.

[33] T. T. Thu and V. Xuan, "Supervised support vector machine in predicting foreign exchange trading," *International Journal of Intelligent Systems and Applications*, vol. 10, pp. 48–56, Sep. 2018.

[34] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[35] M. Ballings, D. Van den Poel, N. Hespeels, and R. Gryp, "Evaluating multiple classifiers for stock price direction prediction," *Expert Systems with Applications*, vol. 42, pp. 7046–7056, 2015.

[36] A. Santuci, E. Sbruzzi, L. Araújo-Filho, and M. Leles, "Evaluation of forex trading strategies based in random forest and support vector machines," *IEEE Latin America Transactions*, vol. 20, p. 2146–2152, 2022.

[37] L. Dongdong, S. Yuan, M. Li, and Y. Xiang, "An empirical study of machine learning algorithms for stock daily trading strategy," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–30, Apr. 2019.

[38] D. Lv, Z. Huang, M. Li, and Y. Xiang, "Selection of the optimal trading model for stock investment in different industries," *PLOS ONE*, vol. 14, p. e0212137, Feb. 2019.

[39] B. Yue, J. Rao, and Y, "A deep learning framework for financial time series using stacked autoencoders and long short term memory," *PLoS ONE*, vol. 12, p. 1–24, 2017.

[40] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, *Deep learning for stock prediction using numerical and textual information*. Okayama, Japan: IEEE, Jun. 2016.

[41] Z. Hu, Y. Zhao, and M. Khushi, "A survey of forex and stock price prediction using deep learning," *Applied System Innovation*, vol. 4, p. 9, Feb. 2021.

[42] H. Jia, "Deep learning algorithm-based financial prediction models," *Complexity*, vol. 2021, p. 5560886, 2021.

[43] J. Sirignano and R. Cont, "Universal features of price formation in financial markets: Perspectives from deep learning," *SSRN Electronic Journal*, vol. 19, p. 1449–1459, 2019.

[44] S. Seifollahi and M. Shajari, "Word sense disambiguation application in sentiment analysis of news headlines: an applied approach to forex market prediction," *Journal of Intelligent Information Systems*, vol. 52, pp. 57–83, 2019.

[45] T. Sprenger, A. Tumasjan, P. Sandner, and I. Welpe, "Tweets and trades: the information content of stock microblogs," *European Financial Management*, vol. 20, pp. 926–957, 2014.

[46] A. Siganos, E. Vagenas-Nanos, and P. Verwijmeren, "Facebook's daily sentiment and international stock markets," *Journal of Economic Behavior &; Organization*, vol. 107, pp. 730–743, Nov. 2014.

[47] R. Dong, M. O'mahony, M. Schaal, K. Mccarthy, and B. Smyth, "Combining similarity and sentiment in opinion mining for product recommendation," *Journal of Intelligent Information Systems*, vol. 46, pp. 285–312, 2016.

[48] P. Joseph and S. Christy, "Improved precision in detecting the early symptoms of covid using k-nearest neighbor and support vector machine algorithm," in *AIP Conference Proceedings*, vol. 2821, AIP Conference Proceedings. AIP Publishing, 2015, p. 080005.

[49] K. Horecki and J. Mazurkiewicz, *Natural Language Processing Methods Used for Automatic Prediction Mechanism of Related Phenomenon*. Cham: Springer International Publishing, 2015.

[50] V. Bijalwan, V. Kumar, P. Kumari, and J. Pascual, "Knn based machine learning approach for text and document mining," *International Journal of Database Theory and Application*, vol. 7, pp. 61–70, Feb. 2014.

[51] A. K. Nassirtoussi, S. Aghabozorgi, T. Y. Wah, and D. Ngo, "Text mining of news-headlines for forex market prediction: A multi-layer dimension reduction algorithm with semantics and sentiment," *Expert Systems with Applications*, vol. 42, pp. 306–324, Jan. 2015.

[52] S. Senarathna and R. Rajapaksha, *Forex Market Prediction Through Text Mining of Financial News*. IEEE, Dec. 2024.

[53] 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, volume 1 (long and short papers)*. 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, p. 4171–4186.

[54] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, A. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[55] J. E. V. Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine learning*, vol. 109, p. 373–440, 2020.

[56] J. M. Duarte and L. Berton, "A review of semi-supervised learning for text classification," *Artificial intelligence review*, vol. 56, p. 9401–9469, 2023.

[57] J. Hao, F. He, F. Ma, and T. Fu, "Trading around the clock: Revisit volatility spillover between crude oil and equity markets in different trading sessions," *Journal of Futures Markets*, vol. 43, p. 771–791, Apr. 2023.

[58] I. Ghosh and P. Dragan, "Can financial stress be anticipated and explained? uncovering the hidden pattern using eemd-lstm, eemd-prophet, and xai methodologies," *Complex & amp; Intelligent Systems*, vol. 9, p. 4169–4193, Dec. 2022.

[59] I. Guyon, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, 2017.

[60] C. J. Neely, D. E. Rapach, J. Tu, and G. Zhou, "Forecasting the equity risk premium: The role of technical indicators," *Institute for Operations Research and the Management Sciences*, 2014.

[61] D. Croce, G. Castellucci, and R. Basili, "Gan-bert: Generative adversarial learning for robust text classification with a bunch of labeled examples," in *Proceedings of the 58th annual meeting of the association for computational linguistics*. Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, p. 2114–2119.

[62] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[63] D. Araci, "Finbert: Financial sentiment analysis with pre-trained language models," *arXiv preprint arXiv:1908.10063*, 2019.

[64] K. K. L. Wong, *The Structure of Neural Network*. Wiley-IEEE Press, 2024, pp. 69–81.

[65] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent neural networks: A comprehensive review of architectures, variants, and applications," *Information*, vol. 15, p. 517, 2024.

[66] F. M. Salem, *Recurrent neural networks*. Springer, 2022.

[67] hochreiter Sepp and schmidhuber jürgen, "Memory, long short-term," *Neural Computation*, vol. 9, p. 1735, 1997.

[68] S. Hochreiter and J. Schmidhuber, "Lstm can solve hard long time lag problems," *Advances in neural information processing systems*, vol. 9, 1996.

[69] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, p. 2451–2471, 2000.

[70] F. A. Gers and E. Schmidhuber, "Lstm recurrent networks learn simple context-free and context-sensitive languages," *IEEE transactions on neural networks*, vol. 12, p. 1333–1340, 2001.

[71] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012.

[72] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint*, p. 1409–1259, 2014.

[73] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, p. 2278–2324, 1998.

[74] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 1–9, 2015.

[75] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *International Conference on Engineering and Technology (ICET)*, p. 1–6, 2017.

Scalable Formal Verification of Modular Concurrent Systems: A Survey of Techniques, Tools and Challenges

Sawsen Khlifa[1], Chiheb Ameur Abid[2], Asma ben Letaifa[3], Belhassen Zouari[4]
Higher School of Communications of Tunis, University of Carthage Ariana, Tunisia[1,3]
Faculty of Sciences of Tunis, University of Tunis El Manar Tunis, Tunisia[2]
LR11TIC05, Mediatron Lab, 2083, Ariana, Tunisia[1,2,3]
ICL-Junia, Université Catholique de Lille, LITL, F-59000 Lille, France[4]

***Abstract*—The increasing complexity of distributed and concurrent systems raises pressing challenges for ensuring correctness and reliability. Formal verification, and in particular model checking, offers a rigorous foundation to validate system properties, yet suffers from the well-known state space explosion problem. This difficulty is especially acute in modular architectures, where local behaviors intertwine with synchronization across components. This paper provides a structured survey of the main techniques designed to overcome these challenges, including state space reduction, abstraction, compositional reasoning, symbolic approaches, and distributed verification. We also review representative tools such as SPIN, NuSMV, LTSmin, DiVinE, and STORM, assessing their capabilities and limitations in handling modular and concurrent models. Building on this landscape, we position the Reduced Distributed State Space (RDSS) as a novel framework that addresses key scalability limits. RDSS reduces global complexity into module-specific metagraphs, ensures stuttering equivalence, and enables local model checking without exploring the full global state space. Comparative experiments demonstrate significant gains over existing approaches, particularly for systems where modules are not all synchronized on the same transitions. We conclude by identifying open challenges and future research directions, including distributed implementations, AI-driven heuristics, and hybrid reductions. Our survey underscores the importance of structural awareness in modern verification workflows and establishes RDSS as a promising foundation for scalable verification of modular concurrent systems.**

***Keywords*—Distributed systems; state space; Modular Petri Net; formal verification; state explosion problem; model checking; temporal logic; reduction techniques; RDSS; ROS2; scalability; modularity; model checking**

I. Introduction

The escalating complexity of modern concurrent and distributed systems has significantly exacerbated the requirement for rigorous verification. From autonomous vehicle networks and smart home infrastructures to critical cloud services and industrial IoT, system components increasingly operate independently, communicate asynchronously, and must coordinate reliably across dynamic environments. This complexity makes ensuring system correctness a central challenge in software engineering.

Because control is distributed and interactions evolve dynamically over time, conventional testing struggles to provide meaningful guarantees about the correctness or robustness of such systems. One effective way to gain confidence in the behavior of such systems is through formal verification. Particularly, model checking stands out as a widely adopted technique. Its automatic, exhaustive nature provides a strong safety net against subtle concurrency bugs. It allows for the exploration of all possible system behaviors to detect design flaws before deployment. However, state space explosion remains a major obstacle [1], [2]. As system components interact, synchronize, and their architectures become more modular, the global state space can grow exponentially, rendering classical model checking intractable, even for moderately sized modular systems.

To mitigate this, a wide array of techniques has emerged, ranging from symbolic representations [3] to partial-order reductions [4], compositional reasoning [5], and on-the-fly verification [6]. These methods, implemented in tools, such as LTSmin [7], SPIN [8], PMC-SOG [9] and NuSMV [10], have significantly improved verification performance. Still, they often struggle when applied to Modular Petri net models, where both local behavior and inter-module synchronizations must be considered. However, the increasing scale and modularity of real-world systems demand approaches that are not only scalable but also structure-aware, able to exploit the modular decomposition of the system to control complexity. Moreover, existing surveys either lack depth, focus on outdated tools, or fail to address the nuances of modular verification in distributed contexts.

In this paper, we propose a structured and up-to-date survey of the techniques and tools used to address the state space explosion in modular and distributed systems.We examine the current landscape of verification tools, identifying those that can support modularity, distributed architectures, and scalable verification. By organizing the literature by technique, tool support, and target application domain, we aim to clarify the strengths and limitations of each class of solution.

We also explore how the formalism of our Reduced Distributed State Space (RDSS) allows for localized verification by isolating module-specific properties expressed in LTL/X. This approach enables efficient model checking that leverages system modularity without requiring global state enumeration.

This paper is intended to guide researchers and practitioners seeking to understand or implement scalable formal verification strategies for distributed systems. Our goal is

to highlight the growing need for structure-aware methods, evaluate the practical impact of recent techniques, and expose gaps in tool support and methodology that merit further investigation.

The structure of this paper is as follows. Section II introduces the background and key concepts underlying formal verification, temporal logics, and Modular Petri nets. Section III analyzes the scalability problem in modular concurrency, detailing the main dimensions of state space explosion. Section IV surveys the principal verification techniques that address these challenges, including reduction methods, abstraction, compositional reasoning, symbolic approaches, and distributed exploration. Section V reviews the tooling ecosystem, highlighting representative model checkers and their capabilities for modular and concurrent systems. Section VI presents the Reduced Distributed State Space (RDSS) framework, outlining its principles, key contributions, and positioning within the literature. Section VII discusses future directions and possible extensions, including distributed implementations, hybrid strategies, and AI-driven heuristics. Finally, Section VIII concludes the paper by summarizing our findings and emphasizing the role of RDSS in advancing scalable verification.

II. Essential Concepts and Preliminaries

Modern concurrent and distributed systems are characterized by the parallel execution of multiple components that often interact asynchronously. To rigorously verify such systems, formal methods —particularly model checking — have become indispensable. This section introduces foundational concepts, including transition systems, temporal logics, Modular Petri nets, and model checking, which underpin many of the techniques discussed in this survey. The above definitions are adapted from Clarke et al. [11] and Baier and Katoen [12], which constitute the classical references in the field of model checking.

A. Labeled Transition System

Labeled Transition Systems (LTS) constitute a fundamental formalism for modeling and analyzing concurrent and distributed systems. These mathematical structures extend the basic concept of transition systems by associating explicit actions with state transitions, thereby enabling a richer representation of dynamic system behavior. LTS finds applications across numerous domains in theoretical computer science, particularly in formal verification, protocol analysis, and reactive system specification. Their capability to distinguish between observable and unobservable actions makes them particularly well-suited for modeling systems where certain internal behaviors must be abstracted from the external interface.

Definition 1: A *Labeled Transition System* (LTS) is formally expressed as a quadruple:

$$LTS = \langle \Gamma, Act, \ \rightarrow, \ I \rangle$$

where:

- Γ denotes a finite ensemble of states,
- Act designates a finite repertoire of actions,
- $\rightarrow \subseteq \Gamma \times Act \times \Gamma$ represents the labeled transition relation,
- $I \subseteq \Gamma$ constitutes the collection of initial states.

For any transition $(s, a, s') \in \rightarrow$, the concise notation

$$s \xrightarrow{a} s'$$

indicates the existence of a passage from state s toward state s', adorned with the label a.

Within the LTS framework, one draws a sharp distinction between:

- Observable actions, symbolized as $Obs \subseteq Act$,
- Unobservable actions, symbolized as $UnObs \subseteq Act$,

such that:

$$Obs \cup UnObs = Act, \qquad Obs \cap UnObs = \varnothing.$$

This partition allows the modeling of two distinct realms: the overtly perceptible behaviors of a system, and its latent, internal evolutions, those hidden pulsations that govern its subterranean logic yet remain invisible to an external observer.

Fig. 1 illustrates LTS, with s_0 serving as the initial state. $\{a, b\}$ make up the set of observed actions. Unlabeled edges are meant to be labeled by non-observed actions.

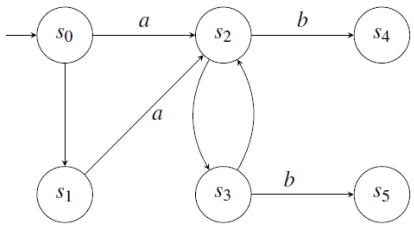


Fig. 1. Example of a labeled transition system.

B. Kripke Structure

In the domain of formal verification, Kripke Structures play a pivotal role as the semantic foundation for temporal logics such as Linear Temporal Logic (LTL) and Computation Tree Logic (CTL). They extend the notion of transition systems by enriching states with atomic propositions that capture observable properties of the system. This labeling mechanism enables the evaluation of temporal formulas, thereby bridging the gap between system executions and logical specifications.

Definition 2: Let AP be a finite set of atomic propositions. A *Kripke Structure* (KS) over AP is a quadruple

$$KS \ = \ \langle \Gamma, \Gamma_0, R, L \rangle,$$

where

- Γ is a finite set of states,

- $\Gamma_0 \subseteq \Gamma$ is the set of initial states,
- $R \subseteq \Gamma \times \Gamma$ is a (total) transition relation,[1] and
- $L : \Gamma \to 2^{AP}$ is a labeling (interpretation) function that assigns to each state the set of atomic propositions true in that state, and each state of the KS is labeled with the values of these propositions.

A Kripke Structure example is shown in Fig. 2. There are two elements in the set of atomic propositions $\{a, b\}$.

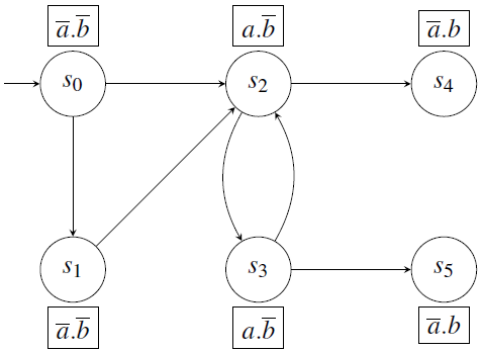


Fig. 2. Example of Kripke Structure.

Paths and reachability. A (finite or infinite) path is a sequence $p = s_0 s_1 s_2 \ldots$ with $(s_i, s_{i+1}) \in R$ for all $i \geq 0$.

- A state $s' \in \Gamma$ is *reachable* from $s \in \Gamma$, written $s \Rightarrow s'$, iff there exists a finite path $s_0 \ldots s_k$ such that $s_0 = s$ and $s_k = s'$.
- A state s is *dead* (a sink) if there is no $s' \in \Gamma$ with $(s, s') \in R$.

Kripke Structures thus furnish the semantic substrate for model checking: a system satisfies a temporal specification precisely when all (or some, depending on the logic) paths of its Kripke Structure satisfy the corresponding formula.

C. Labeled Kripke Structure

While Kripke Structures provide the semantic foundation of temporal logics, in many verification settings it becomes necessary to make explicit the actions that drive state transitions. To address this issue, the notion of a *Labeled Kripke Structure* (LKS) has been introduced. An LKS integrates the action-labeled transitions of a Labeled Transition System (LTS) with the state-labeling of a Kripke Structure (KS). This dual enrichment bridges the gap between transition systems and Kripke semantics, yielding a versatile framework for reasoning simultaneously about state-based properties and action-driven behaviors in concurrent and distributed systems.

Definition 3: Let AP be a finite set of atomic propositions and let Act be a finite set of actions.
A *Labeled Kripke Structure* (LKS) over AP is a quintuple

$$LKS = \langle \Gamma, Act, L, R, \Gamma_0 \rangle,$$

where:

- $\langle \Gamma, Act, R, \Gamma_0 \rangle$ is a Labeled Transition System (LTS),
- $\langle \Gamma, L, R, \Gamma_0 \rangle$ is a Kripke Structure (KS).

An illustration of LKS over $AP = \{a, b\}$ is shown in Fig. 3.

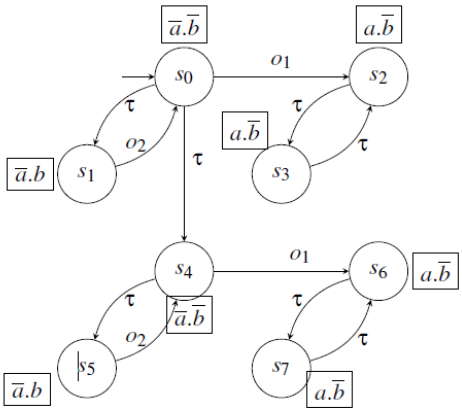


Fig. 3. Example of labeled Kripke Structure.

Remark. An LTS (resp. KS and LKS) can be represented in different ways: explicitly, where each state and arc is stored individually in memory; symbolically, where sets of states are encoded compactly using decision diagrams such as Binary Decision Diagrams (BDDs); or in a hybrid manner, with states represented symbolically while transitions are kept explicit. In addition, such models are often illustrated in an abstract form, for instance through state–transition graphs that emphasize their structural properties rather than their implementation details.

D. Modular Petri Net

Petri nets are a well-established formalism for modeling concurrent and distributed systems. However, when dealing with complex systems, a monolithic representation often becomes intractable. The notion of a Modular Petri Net (MPN) was introduced to address this limitation by decomposing a system into a finite collection of communicating modules, each being an ordinary Petri net. Communication between modules can occur in two ways: A synchronized mode through the fusion of places, or an asynchronous mode through the fusion of transitions. As shown by Christensen and Petrucci [13], any modular net with shared places can be transformed into an equivalent net with only shared transitions. For this reason, in our study we focus exclusively on modules that interact through shared transitions.

Definition 4: A **Modular Petri Net** (MPN) is a pair

$$MPN = (S, TF),$$

that satisfies the following conditions:

1) S is a finite set of **modules** where:
   a) Each module $s \in S$ is an ordinary Petri net

$$s = (P_s, T_s = T_{\text{sync},s} \cup T_{l,s}, W_s, M_{0_s}),$$

[1] Totality means $\forall s \in \Gamma$, $\exists s' \in \Gamma$ such that $(s, s') \in R$.

where, $T_{\text{sync},s}$ denotes the set of **synchronized transitions** of module s, and $T_{l,s}$ denotes the set of **local** (or **internal**) transitions of s.

b) The sets of states and transitions of distinct modules are pairwise disjoint:

$$\forall s_1, s_2 \in S,\ s_1 \neq s_2 \ \Rightarrow\ \begin{array}{l}(P_{s_1} \cup T_{s_1}) \\ \cap\,(P_{s_2} \cup T_{s_2}) = \varnothing.\end{array}$$

c) The global sets of places and transitions are given by

$$P = \bigcup_{s \in S} P_s, \qquad T = \bigcup_{s \in S} T_s.$$

2) $TF \subseteq 2^T \setminus \{\varnothing\}$ is a finite collection of non-empty **transition fusion sets**.

Fig. 4 displays a Modular Petri Net example. The three modules that make up this net are A, B, and C. Both modules A and B are in sync through transition $Sync1$, and both modules B and C are in sync through transition $Sync2$.

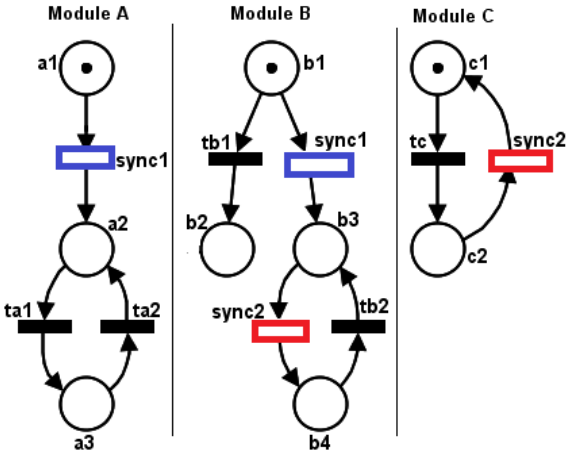


Fig. 4. Example of Modular Petri Net.

Thus, Modular Petri Nets establish a rigorous modeling framework in which complex systems can be decomposed into smaller communicating modules. By distinguishing between local and synchronized transitions, this formalism not only preserves clarity in system representation but also provides a solid foundation for subsequent works that favor modularity and compositional reasoning in the analysis and verification of distributed systems.

E. Temporal Logics: LTL and CTL

Temporal logics are central in the specification and verification of concurrent and distributed systems. They enrich propositional logic with temporal operators, making it possible to express properties not only about individual states but also about the evolution of executions over time. Among the most widely studied are Linear Temporal Logic (LTL) and Computation Tree Logic (CTL), which differ in the way they interpret paths and quantify over possible system behaviors.

1) Linear Temporal Logic (LTL): Linear Temporal Logic (LTL) extends propositional logic with temporal operators to describe both static properties of states and the dynamic progression of system executions. It is widely used to specify properties such as safety (ensuring that undesirable events never occur) and liveness (ensuring that desired events eventually occur).

An LTL formula is built over a finite set AP of atomic propositions that capture the elementary properties of states. Boolean operators ($\neg$, $\wedge$, $\vee$) allow the expression of static conditions, while temporal operators such as X (next), U (until), R (release), F (eventually), and G (always) enable the specification of dynamic properties. Formally, an LTL formula φ is defined over AP, and its models are infinite words over 2^{AP}.

Definition 5: **(LTL Syntax)** The set of LTL formulas is defined inductively as follows:

1) Each $p \in AP$ is an LTL formula.
2) If φ and ψ are LTL formulas, then so are $\neg\varphi$, $\varphi \vee \psi$, $X\varphi$, and $\varphi\, U\, \psi$.

Additional operators are introduced as abbreviations:

$$F\varphi := \top\, U\, \varphi, \qquad G\varphi := \neg F \neg \varphi, \qquad \varphi\, R\, \psi := \neg(\neg\varphi\, U\, \neg\psi).$$

The semantics of LTL link logical formulas to system behaviors by determining whether an execution σ satisfies a formula φ.

Definition 6: **(LTL Semantics)** Let $u = u_0 u_1 u_2 \ldots$ be an infinite word with $u_i \subseteq AP$. For an LTL formula φ, the satisfaction relation $u \models \varphi$ is defined inductively:

- $u \models p$ iff $p \in u_0$,
- $u \models \neg\varphi$ iff $u \not\models \varphi$,
- $u \models \varphi \vee \psi$ iff $u \models \varphi$ or $u \models \psi$,
- $u \models X\varphi$ iff $u_1 u_2 \ldots \models \varphi$,
- $u \models \varphi\, U\, \psi$ iff there exists $k \geq 0$ such that $u_k u_{k+1} \ldots \models \psi$ and for all $0 \leq i < k$, $u_i u_{i+1} \ldots \models \varphi$.

Given a Kripke Structure $K = \langle \Gamma, L, R, s_0 \rangle$, an infinite path $\pi = s_0 \to s_1 \to \ldots$ induces the word $w = L(s_0)L(s_1)\ldots$ over 2^{AP}. We write $K \models \varphi$ if every infinite path π of K satisfies φ.

Remark. The fragment $LTL\backslash X$, which omits the "next" operator, is frequently considered in verification since it guarantees desirable properties such as *stuttering invariance*.

2) Computation Tree Logic (CTL): Computation Tree Logic (CTL) is a branching-time temporal logic, complementary to LTL, that introduces explicit path quantifiers. Whereas LTL formulas are evaluated along a single linear path, CTL formulas are interpreted over the computation tree unfolding from the initial state of a Kripke Structure. Path quantifiers A ("for all paths") and E ("there exists a path") combine with temporal operators X (next), F (eventually), G (globally), and U (until) to form operators such as AX, EF, AG, or $E[\varphi U \psi]$.

Definition 7: **(CTL Syntax)** Let AP be a finite set of atomic propositions. CTL formulas are defined inductively as follows:

1) Each $p \in AP$ is a CTL formula.
2) If φ and ψ are CTL formulas, then so are $\neg\varphi$, $\varphi \vee \psi$.
3) If φ and ψ are CTL formulas, then the following are also CTL formulas: $AX\varphi$, $EX\varphi$, $AF\varphi$, $EF\varphi$, $AG\varphi$, $EG\varphi$, $A[\varphi U \psi]$, and $E[\varphi U \psi]$.

The semantics of CTL formulas are defined with respect to states and their computation trees. Path quantifiers specify whether a property holds across all possible futures (A) or on at least one possible future (E).

Definition 8: **(CTL Semantics)** Given a Kripke Structure $K = \langle \Gamma, L, R, s_0 \rangle$, the semantics of CTL formulas are defined as follows:

- $K, s \models AX\varphi$ iff for all successors s' of s, $K, s' \models \varphi$,
- $K, s \models EX\varphi$ iff there exists a successor s' of s such that $K, s' \models \varphi$,
- $K, s \models AF\varphi$ iff along all paths starting at s, φ eventually holds,
- $K, s \models EG\varphi$ iff there exists a path from s where φ holds globally,
- and analogously for the other operators.

Remark. The essential distinction between LTL and CTL lies in their view of time: LTL formulas are interpreted along single linear paths without explicit quantifiers, while CTL relies on branching-time semantics, combining state formulas with the path quantifiers A and E (see Table I).

TABLE I. COMPARISON BETWEEN LTL AND CTL

Aspect	LTL	CTL
Time model	Linear time: formulas are interpreted along single infinite paths.	Branching time: formulas are interpreted over computation trees with multiple possible futures.
Quantification	Implicit: no explicit path quantifiers, properties are evaluated over all paths.	Explicit: path quantifiers A (all paths) and E (there exists a path) precede temporal operators.
Syntax	Operators: $\neg$, $\vee$, X, U (with derived F, G, R).	Operators: $\neg$, $\vee$, combined with AX, EX, AF, EF, AG, EG, $A[\varphi U \psi]$, $E[\varphi U \psi]$.
Semantics	Evaluated over infinite words induced by paths in a Kripke Structure.	Evaluated over states in a Kripke Structure with respect to their computation tree.
Expressiveness	Suitable for specifying properties of individual executions (e.g. safety, liveness).	Suitable for reasoning about branching choices and existence/universality of behaviors.
Verification	Model checking reduces to automata-theoretic techniques (Büchi automata).	Model checking often based on fixpoint computations over the state space.

F. Model Checking

Model checking is an automated verification technique that, given a finite-state model of a system and a formal property, determines whether the property holds in the model. In the event of a violation, the model checker generates a counterexample, i.e. an execution trace from the initial state to a state that violates the condition. This significantly enhances its practical value, as it helps designers identify and correct errors. Clarke, one of the pioneers of the field, underlined its significance [11]: "It is impossible to overestimate the importance of the counterexample feature. The counterexamples are invaluable in debugging complex systems. Some people use model checking just for this feature."

Based on state-space exploration, the model checking procedure can be described in four main phases [14] (see Fig. 5 [15]):

- State-space generation: Construct the state-space of the model M, representing all possible infinite executions of M. The generated state-space can be modeled as an ω-automaton A_M, whose language $\mathcal{L}(A_M)$ is given in the form of a Kripke Structure.
- Property translation: Transform the LTL property φ into an ω-automaton $A_{\neg\varphi}$, whose language $\mathcal{L}(A_{\neg\varphi})$ consists of all infinite executions that violate φ.
- Product construction: Build the synchronized product automaton $A_M \oplus A_{\neg\varphi}$, which accepts the language $\mathcal{L}(A_M) \cap \mathcal{L}(A_{\neg\varphi})$, representing all executions of M that invalidate φ.
- Emptiness check: Verify whether $\mathcal{L}(A_M) \cap \mathcal{L}(A_{\neg\varphi}) = \varnothing$. If the intersection is empty, the system satisfies φ. Otherwise, the product automaton accepts an infinite word that violates φ, which is then returned as a counterexample.

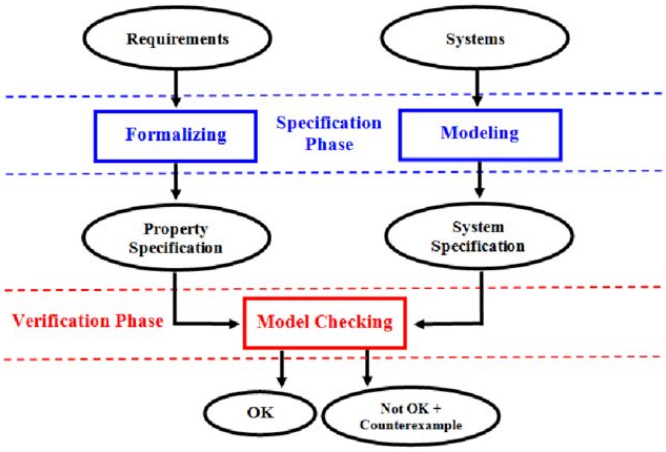


Fig. 5. Schematic view of model-checking approach.

The range of properties addressed by model checking is quite broad. For example, one can verify safety conditions, such as whether a system may reach a deadlock state, or quantitative aspects, such as whether every message will eventually be delivered with a probability of at least 0.99. In both cases, the verification is carried out automatically: the model checker systematically explores the system state space in order to detect any possible violation of the specification.

Several advantages explain the popularity of this technique. It requires little manual effort, as most of the process is

automated; it is versatile, covering a wide variety of correctness properties; and it remains comparatively efficient, even on complex designs. In addition, model checking can be applied even when only partial system specifications are available. While limitations persist, most notably the state explosion problem, the approach has proven to be one of the most practical and widely used methods in the field of formal verification.

However, the practical applicability of model checking is significantly challenged by the state-space explosion problem, which becomes particularly acute in the context of modular and concurrent systems. This limitation motivates the following discussion on scalability issues in modular concurrency.

III. The Scalability Problem in Modular Concurrency

Concurrent and distributed systems, especially when modeled in a modular fashion, pose intrinsic challenges to formal verification. Although model checking provides strong guarantees, its practical use is often hampered by the exponential growth of the state space. This phenomenon is commonly referred to as the scalability problem. In this section, we examine the fundamental dimensions of scalability in modular concurrency, highlighting how structural and semantic aspects of systems trigger combinatorial blow-ups that no straightforward verification method can overcome. These dimensions set the stage for the reduction and abstraction techniques reviewed in the subsequent sections.

A. Dimensions of the Scalability Problem in Modular Concurrency

The scalability problem in modular verification manifests along several complementary dimensions. Each of these dimensions reflects a specific source of combinatorial growth that challenges verification pipelines and limits the applicability of Naïve exploration strategies. By disentangling these dimensions, we gain a clearer view of where reduction and abstraction techniques must intervene to keep verification tractable. In the following, we outline four major dimensions that have been consistently identified in recent research: interleaving blow-up, synchronization hotspots, heterogeneous architectures, and property preservation requirements.

1) Interleaving blow-up: In modular concurrent systems, each independent action multiplies the interleaving possibilities. This leads to a combinatorial explosion of the state space. Even simple system topologies may generate billions of states. Such growth renders explicit exploration infeasible due to memory and time limitations. The issue worsens when components interleave freely, forming dense state graphs that are extremely hard to verify exhaustively.

This challenge has long been identified as a core scalability barrier in model checking [16]. Recent work shows that, even with optimized reductions, state counts can exceed 10^{12} in real-world concurrent systems [17].

2) Synchronization hotspots: Partial-Order Reduction (POR) aims to prune redundant interleavings by merging equivalent execution paths. However, when modules synchronize tightly on shared transitions, independence drops. This reduces the effectiveness of POR, increasing exploration cost. Recent complexity results show that finding near-optimal POR reductions is NP-hard. This underlines the need for hybrid strategies combining POR with abstraction or symmetry to regain scalability [18].

3) Heterogeneous architectures: Modern concurrent systems often execute across heterogeneous platforms such as multi-core CPUs, GPUs, and distributed clusters. Such platforms promise improved scalability. However, they require careful handling of memory consistency to maintain soundness and determinism across parallel or distributed memory. Mismatch in consistency models can cause nondeterministic and incorrect behaviors. Recent work highlights that enforcing cache-coherent shared memory across heterogeneous devices remains challenging, and formal verification must account for these architectural inconsistencies [19].

4) Property preservation: Effective reduction methods must guarantee preservation of critical behavioral properties, especially temporal logic specifications. This usually requires maintaining trace equivalence or stuttering equivalence, essential for logics like LTL without the next-time operator (LTL$\setminus X$). Recent formal work characterizes stuttering equivalence rigorously and reaffirms that LTL$\setminus X$ properties remain valid under such equivalence [20].

To provide a structured overview, Table II summarizes the four core dimensions that shape scalability limits in modular concurrent systems.

TABLE II. Dimensions of the Scalability Problem in Modular Concurrency

Aspect	Description
Interleaving Blow-Up	Exponential growth in interleavings due to concurrent, independent actions.
Synchronization Hotspots	Synchronous transitions undermine Partial-Order Reduction (POR); optimizing POR is NP-hard.
Heterogeneous Architectures	Multi-core, GPU, and cluster deployments challenge soundness and tool integration.
Property Preservation	Reductions must preserve trace or stuttering equivalence to ensure correctness in temporal logics (e.g. LTL\X, CTL*).

B. Real-World Failures Underscoring Scalability Challenges

The scalability issues discussed above are not only theoretical. They have had severe real-world consequences across hardware, avionics, medical systems, and cloud infrastructures.

- Intel Pentium II FDIV bug. A floating-point division error escaped formal analysis and led to costly hardware failures [21].
- Therac-25 radiation accidents. Concurrency-related defects in the control software caused lethal overdoses, revealing verification gaps in safety-critical systems [22].
- Boeing 737 MAX crashes (2018–2019). Insufficient verification of the MCAS flight-control logic under modular and concurrent interactions contributed to catastrophic accidents [23], [24].
- Autonomous vehicle accidents (Tesla, Uber, 2016–2021). Real-time decision modules misinterpreted sen-

sor data due to concurrency, leading to fatal outcomes [25].

- Spectre/Meltdown vulnerabilities (2018). These hardware flaws exposed the limits of verifying speculative execution and microarchitectural concurrency [26].
- Cloud outages (AWS, Azure, Google Cloud, 2020–2022). Complex interactions within distributed services triggered cascading failures that standard verification techniques struggled to predict [28]–[30].

These failures illustrate how state explosion, synchronization complexity, heterogeneous architectures, and property preservation remain key impediments to scalable and trustworthy verification.

IV. Landscape of Scalable Verification Techniques

A wide range of techniques has been proposed to mitigate the scalability challenges of model checking.

As shown in Fig. 6, these approaches can be organized into a taxonomy that highlights the main methodological categories, which will be detailed in the subsequent subsections.

These approaches range from algorithmic reductions that shrink the explored state space, to compositional reasoning frameworks that exploit modularity, and to strategies that harness high-performance computing platforms such as multi-core processors, GPUs, and distributed clusters. This section provides a structured overview of the methodological principles that underpin scalable verification. We highlight algorithmic strategies like state space reduction, abstraction, compositional reasoning, distributed exploration, and symbolic approaches, which form the foundation for advancing model checking in modular and concurrent systems.

A. State Space Reduction Methods

1) Partial-Order Reduction (POR): Partial-Order Reduction (POR) reduces redundant interleavings by exploiting the independence of concurrent actions. It has become one of the most effective techniques to alleviate the state explosion problem in modular and concurrent systems.

A quasi-optimal POR framework was formalized in [31], demonstrating how near-optimal reductions can drastically shrink the explored space in realistic models. Practical advances have followed, such as context-sensitive DPOR with observers, which improves scalability while preserving correctness [32]. Other recent work proposes interference-based DPOR to further prune unnecessary interleavings [33]. At the same time, theoretical results confirm the intrinsic hardness of optimizing POR, proving that even approximate solutions remain NP-hard [18].

These developments underline both the practical benefits of POR in real verification pipelines and the fundamental complexity limits that still motivate new heuristics and hybrid strategies.

2) Symmetry reduction: Symmetry reduction (SR) addresses the state explosion problem by identifying and merging states that are equivalent under permutations of identical components. For instance, in systems composed of many replicated processes, the order in which two identical processes execute is often irrelevant; exploring only one such case is sufficient.

This principle has been successfully applied in the context of Communicating Sequential Processes (CSP), a process algebra for modeling concurrent systems. Using the Failures–Divergences Refinement (FDR) model checker, groups of symmetric states are collapsed so that only a single representative is explored. This technique has enabled the verification of models with otherwise extremely large state spaces [34].

More recently, for distributed round-based protocols, partition symmetry reduction and message-order reduction dramatically shrink the model size by exploiting template symmetries [35].

Another promising direction is the use of counter abstraction, where systems with many identical components are represented by counting processes instead of enumerating each explicitly. The recent work of Eichler et al. [36] introduces a precise $(0, 1)$-counter abstraction for parameterized verification. This approach leverages process symmetry while preserving essential behavioral distinctions, thus enabling scalable verification of modular concurrent systems with large numbers of replicated components.

Together, these results illustrate not just how leveraging symmetry enables scalability in modular settings, but also how such reductions can be integrated into modern verification frameworks.

3) Stubborn sets: Stubborn sets are a state space reduction method that, like partial-order reduction, exploits the independence of concurrent actions. The idea is to identify a subset of transitions as the stubborn set that suffices to represent all relevant behaviors from a given state. By exploring only these transitions, large portions of equivalent interleavings can be safely omitted while preserving essential properties, like reachability and stuttering-invariant temporal logic formulas [LTL\X).

The core principle of this method continues to be refined and adapted to new and complex verification scenarios. For example, it has been extended to handle two-player reachability games, providing efficient reduction in adversarial settings where the outcome depends on the choices of multiple agents [37].

Furthermore, the versatility of stubborn sets is demonstrated by their application beyond standard Petri nets; they have been successfully integrated into frameworks for verifying Time Petri Nets, helping to mitigate state explosion in real-time systems by reducing redundant interleavings of temporal actions [38].

These developments underscore that stubborn sets remain a fundamental and adaptable technique for tackling state space explosion in diverse types of concurrent systems.

B. Abstract State Space Representations

Abstraction is a fundamental method to tackle the state explosion problem by collapsing sets of concrete states into

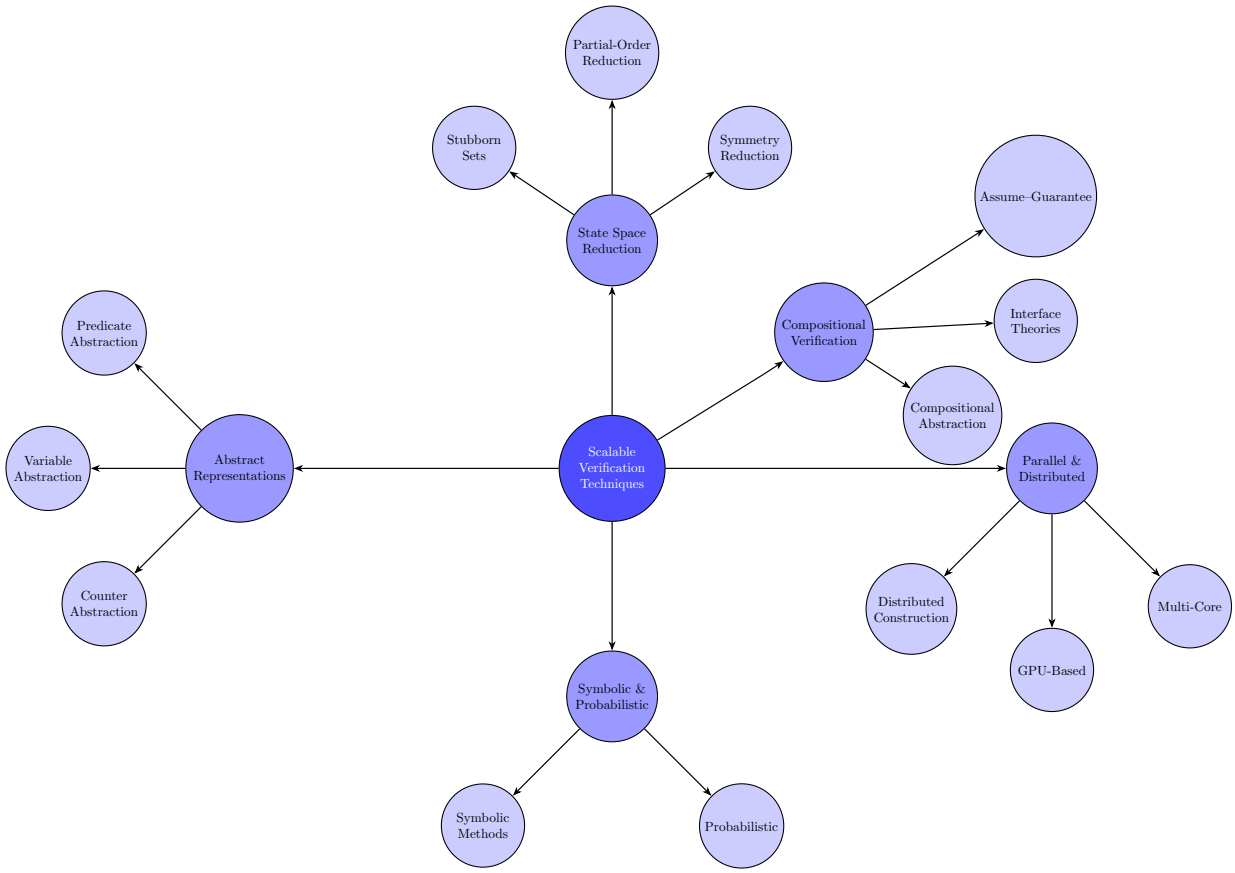


Fig. 6. Overview of scalable verification techniques, categorized into state space reduction, abstraction, compositional reasoning, parallel and distributed approaches, and symbolic/probabilistic methods.

abstract representatives. In contrast to reductions such as POR or symmetry, which preserve the concrete structure but prune interleavings, abstraction constructs a higher-level model that approximates the original system. Three categories of abstraction are commonly distinguished in the literature:

1) Predicate abstraction: Predicate abstraction replaces concrete data domains with Boolean predicates that capture properties relevant to the verification task. This technique underpins the well-known CEGAR (Counterexample-Guided Abstraction Refinement) framework, where spurious counterexamples trigger automatic refinement. Recent work applies predicate abstraction to real-time multi-agent systems, producing finite abstract models that remain correct for temporal logics, such as TCTL [39].

2) Data / Variable abstraction: Variable abstraction reduces large or infinite data domains into coarser partitions such as intervals, symbolic ranges, or sets. This approach is particularly useful in systems with timing or quantitative data, where abstraction preserves the control structure but simplifies the analysis. Examples include abstractions used for timed automata and hybrid systems, where compact representations allow tractable model checking [40].

3) Counter abstraction: Counter abstraction (also known as symmetry-based abstraction) aggregates identical processes by recording how many are in each local state, rather than tracking each process individually. This method has been widely used in the parameterized verification of concurrent protocols, significantly reducing the state space of replicated-component systems. Recent advances refine counter abstraction with precise $(0, 1)$-counters for parameterized systems, ensuring scalability without losing critical behaviors [36].

These three categories form the canonical core of abstraction techniques in model checking, offering scalable verification through higher-level representations of system behavior.

C. Compositional Verification

Compositional verification is a fundamental strategy to address state explosion by leveraging the modular structure of a system. Instead of verifying the system monolithically, this approach decomposes the verification task by analyzing individual components in isolation. The results of these local analyses are then composed to infer global properties of the entire system. The core challenge lies in managing the interactions between components, typically through the use of assumptions, contracts, or abstract interfaces.

1) Assume-guarantee reasoning: Assume-Guarantee Reasoning (AGR) is a cornerstone compositional technique that verifies systems by analyzing their parts separately. Instead of building a single large model for the entire system, AGR checks each component under specific assumptions about how

others will behave. The core idea is captured by a simple rule: a component C guarantees a property G provided its environment meets an assumption A, written as $\langle A \rangle C \langle G \rangle$.

To verify a system made of two components, suitable assumptions must be found so that each component's guarantee holds when the other acts as assumed. This avoids constructing the global state space altogether. The main challenge is automatically discovering these assumptions.

Recent advances are expanding the scope and expressiveness of AGR along two key dimensions: by developing novel logical frameworks to specify complex compositional properties, such as using epistemic logic to reason about knowledge in concurrent systems [5], and by creating powerful symbolic techniques that apply this reasoning to verify intricate systems, such as multi-language security protocols [27].

2) Interface theories: Interface Theories provide a formal foundation for modeling component interactions through contracts. A contract explicitly defines the assumptions a component makes on its environment and the guarantees it provides under those conditions. This formalism enables rigorous compositional reasoning by ensuring that each component's guarantees satisfy the assumptions of others in the system, thereby guaranteeing overall correctness [41].

Contract-Based Design (CBD) elevates this concept to a systems engineering methodology. It promotes modularity, reuse, and incremental development by allowing components to be specified, verified, and refined independently based on their contracts. This approach is particularly effective for large-scale concurrent systems, as it localizes verification efforts and simplifies integration. Recent research has focused on overcoming the practical challenge of manual contract specification by developing methods for the automated composition and refinement of contracts, making the methodology more scalable and accessible for complex system-on-chip designs [42]. The practical impact of CBD is significant, with successful applications spanning critical domains such as cyber-physical systems (CPS), where it helps manage the complexity of interactions between computational and physical elements [43]. This also extends to the automotive and aerospace industries.

3) Compositional abstraction and minimization: This technique involves generating a simplified, abstract model for each component that is conservative with respect to the properties of interest. The key idea is to reduce the state space of individual components before composition, making the subsequent global analysis more tractable. The abstraction must be precise enough to preserve relevant behaviors when components interact.

Methods range from traditional abstraction refinement (CEGAR) applied compositionally to techniques that exploit the modular structure to minimize each component while preserving equivalence (e.g. bisimulation minimization).

A recent advancement in this area focuses on compositional model checking for multi-properties, which aims to verify multiple system properties simultaneously and efficiently by leveraging the compositional structure [44]. The final verification step is performed on the composition of the abstract models, which is typically much smaller than the composition of the original components.

D. Distributed and Parallel Verification

The exponential growth in state space size often exceeds the capabilities of sequential verification, necessitating approaches that leverage parallel and distributed computing architectures. This subsection explores three key strategies that exploit modern hardware to distribute the computational and memory load of verification. These techniques, which target multi-core CPUs, GPUs, and distributed clusters respectively, form an essential toolkit for scaling verification to extremely large and complex modular systems.

1) Multi-core verification: A key strategy to combat state explosion leverages the parallel processing power of multi-core CPUs. This involves designing model checking algorithms and data structures that can efficiently distribute the workload across multiple cores, aiming for near-linear speedups. The challenge lies not only in parallelizing state space exploration but also in ensuring that the underlying data structures scale without becoming a bottleneck due to memory contention or fragmentation.

Recent innovations address this core challenge head-on. The EDDY framework exemplifies this progress, introducing a novel multi-core Binary Decision Diagram (BDD) package specifically engineered for modern shared-memory architectures [45]. Its main contributions include advanced dynamic memory management and a fragmentation-reducing allocator, which together mitigate critical performance overheads that plague traditional implementations. By optimizing memory access patterns and locality, EDDY enables more efficient parallel symbolic verification, directly increasing the scale of systems that can be checked. This focus on low-level memory optimization complements higher-level concurrency strategies, such as work stealing and lock-free data structures, forming a robust and scalable foundation for multi-core model checking.

a) GPU-based model checking: The massive parallelism offered by GPUs is exploited to accelerate demanding verification tasks, particularly for complex system models. This approach moves beyond traditional CPU-bound algorithms to harness the thousands of cores available on a GPU for parallel state space exploration and analysis.

GPU acceleration effectively scales Statistical Model Checking (SMC), a key method for verifying stochastic and timed systems. For instance, executing simulations of Extended Timed Automata on the GPU significantly accelerates the verification of temporal and probabilistic properties [46]. This approach efficiently tackles state space explosion in complex real-time models.

2) Distributed state space construction: Distributed verification decomposes the global state space across a network of machines, each responsible for exploring a distinct partition. This approach directly addresses the memory limitations of a single node, enabling the analysis of exceptionally large systems. The performance and scalability of this strategy critically depend on the underlying network technology and the efficiency of state management.

Cutting-edge research now leverages high-speed RDMA networks to minimize communication overhead. For instance, recent algorithms combine distributed exploration with sophisticated state reconstruction techniques, dramatically reducing

the amount of data that needs to be transmitted between nodes. An evaluation of such an approach shows significant improvements in scalability and efficiency for explicit state model checking, demonstrating the potential of modern networking technologies for distributed verification [47]. This line of work is directly applicable to large-scale modular systems, including those using RDSS-style construction.

E. Symbolic and Probabilistic Verification

While the previous techniques often deal with the state space explicitly or by exploiting its structure, another family of approaches uses fundamentally different representations. This part covers two such powerful paradigms: symbolic verification, which uses compact encodings to represent vast state spaces implicitly, and probabilistic verification, which extends classical model checking to reason about stochastic behaviors and quantitative properties. The integration of symbolic and probabilistic methods opens new frontiers for scalable verification.

1) Symbolic methods: Symbolic model checking techniques avoid explicit state enumeration by using compact logical representations. These methods employ data structures such as Binary Decision Diagrams (BDDs) and satisfiability solvers (SAT/SMT) to encode state sets and transition relations as formulas. This approach enables efficient property checking by exploring large state spaces symbolically rather than explicitly.

The practical effectiveness of these techniques depends fundamentally on the performance of their underlying reasoning engines. Modern SAT/SMT solvers have demonstrated remarkable capability in solving complex constraint problems across various domains. As shown in recent evaluations of large-scale puzzle solving, these solvers can efficiently handle problems with millions of constraints and variables, showcasing their scalability and robustness [48]. In formal verification, this computational power enables the analysis of complex system properties by reducing verification tasks to satisfiability problems, making symbolic methods essential for handling large-scale industrial systems.

2) Probabilistic verification: Probabilistic model checking extends classical verification to systems that exhibit random or unpredictable behavior. This approach analyzes quantitative properties of models such as Markov chains, which describe systems evolving probabilistically over time. Unlike traditional verification that provides yes/no answers, probabilistic methods compute likelihoods, for example, determining the probability that a communication protocol successfully delivers a message or calculating the expected energy consumption of a distributed algorithm.

The theoretical foundation for efficiently analyzing these models relies heavily on notions of behavioral equivalence. Recent advances have focused on developing a spectrum of approximate probabilistic bisimulations, which provide a formal framework for quantifying the similarity between states in probabilistic systems [49]. This allows for the construction of simplified models that preserve quantitative properties within precise error bounds, enabling more efficient analysis.

To consolidate this discussion, Table III provides a structured summary of the main categories of scalable verification techniques, their key methodological principles, typical domains of application, and their suitability for modular concurrency.

Remark. Some references appear in multiple sections of this survey. This reflects their dual relevance: for instance, a contribution may introduce a novel reduction technique while also providing its implementation in a widely used tool. In such cases, the same source is cited in both the methodological and tooling contexts to accurately represent its impact across the landscape.

V. Tooling Ecosystem: Capabilities for Modular Systems

In parallel with theoretical advances, a rich ecosystem of verification tools has been developed. These platforms implement the reduction, abstraction, and compositional techniques surveyed in the previous section, thus bridging academic research with industrial applications. This section reviews the most influential verification tools and evaluates their relevance for modular and concurrent systems.

A. Classical Explicit-State Tools

Classical explicit-state tools represent the first generation of practical model checkers. They operate by directly constructing and exploring the state space. They are often supported by sophisticated search strategies and reduction techniques to manage the explosion of states. Although more recent symbolic and distributed approaches have emerged, these tools remain highly influential. They are still actively used in both academia and industry. The following subsections briefly review three representative explicit-state platforms.

1) SPIN: The SPIN model checker [8] remains one of the most widely used explicit-state verification engines. It targets systems specified in the Promela modeling language and implements techniques such as partial-order reduction and bitstate hashing. SPIN has been applied to a broad spectrum of communication protocols and distributed algorithms, demonstrating scalability up to millions of states. While not natively modular, its support for asynchronous processes makes it relevant for analyzing concurrent behaviors in modular settings.

2) NuSMV: NuSMV [10] is a symbolic model checker supporting both CTL and LTL properties. It implements BDD-based symbolic methods and SAT-based bounded model checking, providing a balance between explicit and symbolic exploration. Although its architecture is not specifically modular, NuSMV has served as a foundation for contract-based verification frameworks, notably in embedded and safety-critical software.

3) UPPAAL: UPPAAL [50], [51] focuses on timed automata and real-time systems. It integrates symbolic techniques for handling clock variables, making it particularly suited for modular cyber-physical systems. Compositional extensions of UPPAAL support verification of large networks of timed components, where timing constraints play a central role.

B. High-Performance Verification Frameworks

High-performance frameworks emerged to overcome the scalability limitations of classical explicit-state tools. These

TABLE III. COMPARATIVE SUMMARY OF SCALABLE VERIFICATION TECHNIQUES, THEIR DOMAINS, AND SUITABILITY FOR MODULAR CONCURRENCY

Category	Key Methods / References	Domain of Application	Suitability for Modular Concurrency
State Space Reduction	POR [18], [31]–[33], Stubborn Sets [37], [38], Symmetry [35], [36]	Protocols, Hardware verification, Petri nets	High — directly addresses interleaving and replication in concurrent modules
Abstract Representations	Predicate [39], Variable/Data [40], Counter [36]	Hybrid systems, Real-time models, Parameterized protocols	Medium to High — effective when abstraction is precise, but risk of spurious behaviors
Compositional Verification	AGR [5], [27], Interface Theories [41]–[43], Compositional Abstraction [44]	Large-scale software, Cyber-physical systems, Security protocols	High — inherently modular, though assumption discovery and contract design remain challenging
Parallel & Distributed, Symbolic & Probabilistic Verification	Multi-core [45], GPU [46], Distributed [47], Symbolic/Probabilistic [48], [49]	Industrial-scale models, Stochastic protocols, Cloud infrastructures	Medium to High — excellent for scaling exploration, less natural for modular decomposition

platforms exploit optimized data structures, parallel architectures, and language-independent interfaces to accelerate state space exploration. They provide the backbone for verifying larger and more complex models, while often serving as integration points for new reduction techniques and symbolic back-ends. The following subsections review representative frameworks in this category.

1) LTSmin: LTSmin [52] is a high-performance language-independent model checker that provides interfaces to multiple modeling languages (e.g. Promela, DVE, PNML). It integrates symbolic techniques (decision diagrams), parallel reachability, and partial-order reductions. Its modular architecture and native support for Petri nets make it highly relevant for modular concurrency.

2) DiVinE: DiVinE [53] is a distributed explicit-state model checker designed for large-scale verification on clusters and high-performance computing platforms. It implements parallel reachability, on-the-fly LTL model checking, and advanced partial-order reductions. DiVinE demonstrates strong scalability for concurrent modular models distributed across thousands of cores.

3) EDDY (multi-core BDD): The EDDY framework [45] provides a multi-core BDD package optimized for shared-memory architectures. By redesigning memory allocators and dynamic variable reordering, EDDY improves symbolic model checking on multi-core CPUs. This directly addresses the need for efficient symbolic verification in modular systems with large state spaces.

C. Probabilistic and Symbolic Engines

Another important branch of verification frameworks focuses on symbolic and probabilistic approaches. Instead of relying solely on explicit state enumeration, these engines use compact logical representations or stochastic models to capture system behaviors. Symbolic methods such as BDD- or SAT/SMT-based encodings enable implicit exploration of vast state spaces, while probabilistic model checkers extend classical verification to reason about uncertainty and quantitative properties. Together, these tools greatly expand the range of systems and properties that can be analyzed. The following subsections highlight representative engines in this category.

1) STORM: STORM is a modern probabilistic model checker that integrates symbolic, explicit, and numerical methods into a unified framework. It supports a wide range of probabilistic models, including Markov decision processes (MDPs), continuous-time Markov chains (CTMCs), and stochastic Petri nets, making it applicable to both academic benchmarks and industrial-scale case studies. Unlike earlier tools that specialized in either symbolic or explicit techniques, STORM provides a flexible architecture that combines multiple engines to balance scalability and precision. Recent evaluations confirm its ability to analyze large and complex quantitative models while maintaining soundness and efficiency [54]. This makes STORM particularly relevant for modular and concurrent systems where stochastic behaviors interact with structural concurrency.

2) PRISM: PRISM [55] pioneered probabilistic model checking and remains a widely used tool for analyzing stochastic protocols and cyber-physical systems. It supports a wide range of probabilistic logics and has influenced the design of STORM. While not primarily modular, its modeling language facilitates component-based specifications.

3) SAT/SMT-based frameworks: Modern SAT/SMT-based model checkers leverage the efficiency of constraint solvers to perform bounded or unbounded verification. Tools such as CBMC and IC3-based solvers integrate abstraction-refinement and compositional reasoning, making them suitable for modular concurrent software verification. Recent advances demonstrate their scalability for verifying security protocols and hardware blocks.

D. Domain-Specific and Hybrid Tools

Beyond general-purpose model checkers, a variety of domain-specific and hybrid tools have been developed to address the needs of particular application areas. These platforms often combine multiple verification paradigms (explicit, symbolic, probabilistic) and tailor them to specialized modeling languages or industrial domains. They play a crucial role in bridging the gap between theoretical advances and real-world deployments, offering verification capabilities that are more directly aligned with practitioner requirements. The following parts survey representative tools in this category.

1) PAT (Process analysis toolkit): PAT [56] supports verification of process-algebraic models, combining CSP-style semantics with state space reductions and refinement checking. It provides explicit support for modular specifications and has been applied to distributed protocols and workflow systems.

2) SDV (Static driver verifier): Microsoft's Static Driver Verifier (SDV) integrates symbolic model checking with static analysis to automatically verify Windows device drivers [57]. It

builds on the SLAM project and uses predicate abstraction with refinement. Although domain-specific, SDV exemplifies how verification pipelines can scale to real-world modular software components.

3) Recent hybrid platforms (GPU, cloud): Recent tools leverage GPUs and cloud infrastructures to accelerate verification. Notably, distributed model checkers such as DiVinE-cloud and GPU-accelerated statistical model checking frameworks demonstrate promising results [46], [47]. These platforms highlight the emerging trend of integrating heterogeneous computing into verification workflows.

4) SPORE: SPORE [58] is a recent stateless model checker that combines partial-order reduction with symmetry reduction. Its hybrid algorithm significantly reduces explored interleavings, demonstrating orders-of-magnitude improvements over classical POR. SPORE directly targets concurrency and is particularly relevant for modular distributed protocols. Although some of these tools were introduced decades ago, they continue to be actively maintained and widely applied in recent studies.

Table IV summarizes their core verification techniques, supported models, relevance for modular/concurrent systems, and illustrates their continued impact with recent references and application domains.

While many of these tools were originally introduced more than a decade ago, their sustained use in recent research highlights the robustness of their foundational techniques. As shown in Table IV, extensions such as GPU acceleration, symbolic encodings, and domain-specific adaptations (e.g. robotics, IoT, autonomous driving) ensure that these platforms remain relevant for contemporary verification challenges. This consolidated landscape provides the backdrop against which we now situate our own contribution, focusing on the Reduced Distributed State Space (RDSS) framework and its role in addressing scalability in modular concurrent systems.

VI. RDSS: Contribution, Positioning, and Future Directions

Building upon the surveyed techniques and tools, this section introduces the Reduced Distributed State Space (RDSS) framework as a novel approach to scalable verification of modular and concurrent systems. RDSS directly addresses the limitations identified in existing methods by providing a modular structure that integrates reduction principles while enabling verification at the level of individual modules. We first outline the gaps in current approaches, then present the RDSS framework and its key contributions, followed by a discussion of its positioning within the literature and possible future extensions.

A. Scope and Claims

Before detailing our framework, we explicitly clarify its scope and novelty. The novel contribution of RDSS lies in its ability to perform local model checking directly on per-module meta-graphs, rather than requiring construction of the entire global state space. This is achieved through two key mechanisms: (*i*) τ-hiding of irrelevant synchronizations, and (*ii*) meta-state fusion that preserves *stuttering equivalence*. These innovations enable verification of module-specific properties with strong theoretical guarantees. RDSS does not reinvent prior reduction strategies; instead, it *integrates* and benefits from established methods such as partial-order reduction, symmetry reduction, and abstraction, while remaining compatible with distributed exploration. Finally, the property classes supported are those expressible in the stutter-invariant fragment of LTL (LTL $\setminus$ $\mathbf{X}$), including safety, liveness, and local properties such as deadlock-freedom, home-space reachability, and module-level liveness. By stating these boundaries, we aim to separate our claims from the survey of existing work and set clear expectations for the reader.

B. Identified Gaps in Existing Approaches

Despite the advancements in model checking and state space reduction methods, current verification techniques for modular and concurrent systems face significant scalability challenges. Traditional approaches, such as POR, symmetry reduction, and abstraction, address specific aspects of the state space explosion problem. Unfortunately, they do not provide comprehensive solutions for the complex nature of modern systems.

First, POR helps reduce redundant interleavings by exploiting the independence of actions. However, its effectiveness is limited when the system's concurrency is highly interdependent or exhibits complex synchronization. Symmetry reduction identifies equivalent states that result from symmetric components. Nevertheless, it fails to address cases where the symmetry is less evident or dynamic. Similarly, abstraction techniques can collapse large portions of the state space. However, they risk oversimplifying behaviors, potentially missing critical verification information. This is particularly problematic when applied to modular systems with complex interactions between components.

Moreover, these techniques, while effective individually, fail to handle the full scope of challenges posed by highly dynamic and modular systems, where components can interact in unpredictable ways. These limitations motivate the need for more integrated and scalable approaches, such as Reduced Distributed State Space (RDSS), which tackles the scalability problem in a modular and distributed context.

C. The RDSS Framework

The RDSS framework introduces a novel and scalable solution to the state explosion problem in modular and concurrent systems. It builds upon the distributed state space structure originally proposed in [80], extending it with formal guarantees such as stuttering equivalence, τ-hiding of irrelevant synchronizations, and fusion of equivalent meta-states. This part first presents the core principles of RDSS, followed by its key contributions to scalable verification.

1) The RDSS principle: The RDSS framework addresses the challenge of state explosion in modular systems by reducing the global state space into smaller, manageable meta-graphs formed of meta-states that represent the state space of individual modules. The input of the RDSS consists of the concurrent modular system specification being studied, formalized in a Modular Petri Net structure. Each module in the system constructs its local meta-graph, capturing only the relevant transitions based on local and synchronized events.

TABLE IV. SUMMARY OF REPRESENTATIVE VERIFICATION TOOLS, THEIR CORE TECHNIQUES, SUPPORTED MODELS, RELEVANCE FOR MODULAR/CONCURRENT SYSTEMS, AND EVIDENCE OF CONTINUED USE IN RECENT DOMAINS

Tool	Core Techniques	Model Supported	Relevance to Modular/Concurrent Systems	Recent Use (Refs.)	Application Domain (Recent)
SPIN	Explicit-state, POR, Bitstate hashing	Promela processes	Asynchronous processes, distributed protocols	[59], [60]	IoT-based transport, abstracted IoT protocols
NuSMV	Symbolic (BDD, SAT/SMT), Bounded MC	SMV language, CTL/LTL	Contract-based reasoning in modular embedded systems	[61], [62]	Multi-agent systems, software product lines
UPPAAL	Symbolic (timed automata, DBMs)	Networks of Timed Automata	Real-time CPS and IoT verification	[63], [64]	Drone access control, contract automata runtime
LTSmin	Symbolic DDs, POR, Distributed exploration	PNML, Promela, mCRL2	Modular Petri Nets and parameterized models	[65], [66]	GPU-based LTL MC, knowledge-based reductions
DiVinE	Explicit distributed MC, Parallel LTL checking	DVE, PNML, Promela	Cluster-based concurrency analysis	[67], [68]	C++ program verification, GPU POR
EDDY	Multi-core symbolic BDDs	Symbolic encodings (BDD-based)	Shared-memory multi-core verification	[69], [70]	HPC memory optimization, GPU model training
STORM	Symbolic + numerical probabilistic engines	MDPs, CTMCs, Stochastic PN	Quantitative verification of stochastic modular systems	[71], [72]	Robotic systems, MDP algorithm selection
PRISM	Symbolic (BDD, MTBDD), Probabilistic checking	Probabilistic automata, MDP, CTMC	Stochastic CPS, security protocols	[73], [74]	Symmetry reduction, model-based frameworks
SPORE	Stateless MC, POR + Symmetry reduction	Concurrent C programs, message-passing protocols	Concurrency, modular distributed protocols	[58]	Benchmarks of concurrency programs
PAT	Explicit + refinement checking, POR	CSP-style models, workflows	Modular process-based verification	[75], [76]	Federated learning, blockchain smart contracts
SDV	Predicate abstraction + CEGAR (SLAM)	C code (Windows drivers)	Modular verification of device drivers	[77], [78]	Distributed BMC, autonomous driving (ADS)

The interactions between modules in RDSS are captured through synchronization points, which align the meta-graphs of different modules during the construction process. This modular approach avoids the need to explore redundant state space information from non-relevant modules.

A key concept in RDSS is the use of τ transitions, which are used to hide irrelevant synchronizations between modules. These transitions allow the system to focus on the core behaviors, ignoring unnecessary synchronization events that do not affect the properties being verified. During the construction of the RDSS, equivalent meta-states are identified and merged to reduce the overall state space. This process ensures that only unique behaviors are captured, and redundant states that exhibit identical or analogous transitions are eliminated.

This fusion is based on the principle that multiple states, which share the same output behavior and synchronization properties, can be represented by a single meta-state. By merging these equivalent meta-states, RDSS maintains the correctness of the verification while significantly reducing the number of states that need to be explored [81].

The local meta-graphs are then used for verification, enabling the system to scale by performing module-specific verification without requiring access to the entire global state space [79].

Furthermore, RDSS ensures the stuttering equivalence of the reduced state space. It has been formally proven that the projection of the language generated by the graph of a module on its own transitions is the same as the language generated by the entire system's global state space. This guarantees that no critical behaviors are lost during the reduction process.

Fig. 7 demonstrates the modular decomposition and the use of synchronization points in the RDSS framework.

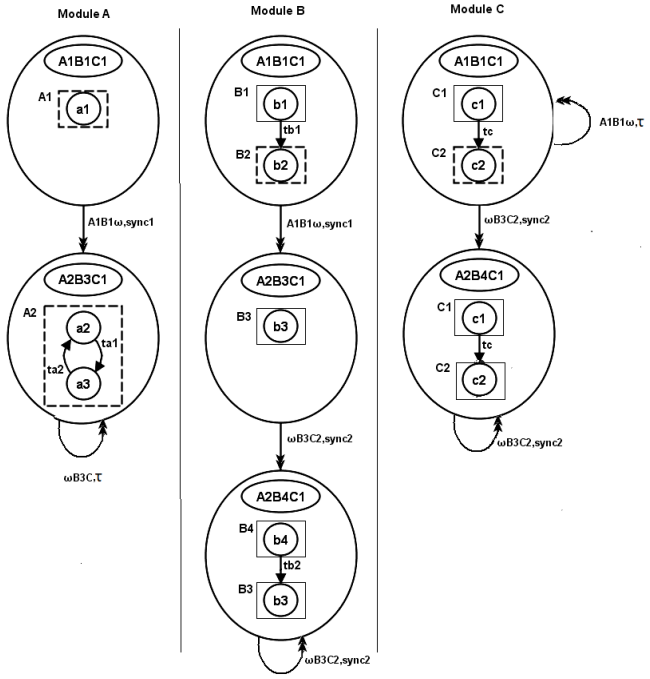


Fig. 7. Illustration of RDSS corresponding to the Modular Petri Net of Fig. 4.

D. Soundness of Local Model Checking in RDSS

A central question for the Reduced Distributed State Space (RDSS) framework is whether properties verified locally on module-level meta-graphs are sound with respect to the original Modular Petri net (MPnet). This subsection establishes the formal basis of this claim by proving that RDSS preserves stuttering equivalence with the original system, and that the projection of behaviors on a given module coincides in both

structures. As a result, $LTL \setminus \mathbf{X}$ properties can be checked locally without loss of correctness.

1) Projection argument: Let Σ be the language generated by the MPnet $\mathcal{P}$ over its set of transitions, and $\Sigma_{/s}$ the projection of Σ onto the transitions specific to module s. Let Σ_s be the language generated by the meta-graph of module s, and $\Sigma_{s/s}$ its projection onto the transitions of s. We prove that $\Sigma_{/s} = \Sigma_{s/s}$.

Indeed, any infinite execution sequence $\sigma \in \Sigma$ can be written as

$$\sigma = \left((t_l)^*(t_{sync})^*\tau^*\right)^*,$$

where, t_l are local transitions, t_{sync} are synchronized transitions, and τ denotes hidden synchronizations. Projecting σ onto module s eliminates irrelevant synchronizations, yielding

$$\sigma^{\Sigma_{/s}} = \left((t_{l,s})^*(t_{sync,s})^*\right)^* \in \Sigma_s.$$

Hence $\Sigma_{/s} \subseteq \Sigma_{s/s}$.

Conversely, every execution $\sigma \in \Sigma_s$ is built from the local, synchronized, and τ transitions of module s, so

$$\sigma = \left((t_{l,s})^*(t_{sync,s})^*\tau^*\right)^*.$$

Projecting σ on s removes the τ transitions, giving

$$\sigma^{\Sigma_{s/s}} = \left((t_{l,s})^*(t_{sync,s})^*\right)^* \in \Sigma_{/s}.$$

Thus $\Sigma_{s/s} \subseteq \Sigma_{/s}$. Combining both inclusions yields:

$$\Sigma_{/s} = \Sigma_{s/s}.$$

2) Stuttering equivalence: From the above projection result, we conclude that the maximal paths of the MPnet and those of the RDSS coincide modulo τ-hiding. This implies that the two structures are stuttering-equivalent, and therefore preserve all properties expressible in the stutter-invariant fragment $LTL \setminus \mathbf{X}$.

Theorem 1 (Stuttering Equivalence for RDSS): Let $\mathcal{P}$ be an MPnet with $\mathcal{P} = (S, TF)$, and let $\mathcal{R}$ be its corresponding RDSS defined as

$$\mathcal{R} = \{\, RG_s = (\widehat{\mathcal{N}}_s, \widehat{\mathcal{A}}_s) \mid s \in S \,\}.$$

For any $LTL \setminus \mathbf{X}$ formula φ defined over T_s, the set of local and synchronized transitions of a module s, we have:

$$\mathcal{P} \vDash \varphi \iff \mathcal{R} \vDash \varphi.$$

3) Proof sketch: The projection argument shows that every path of $\mathcal{P}$ restricted to module s corresponds to a path in $\mathcal{R}$ and vice versa. Since $LTL \setminus \mathbf{X}$ properties are preserved under stuttering equivalence, verifying φ on the module-level meta-graph is sound and complete with respect to verifying it on the global reachability graph. Detailed proofs and demonstrations are available in our GitHub repository[2].

4) Key contributions of RDSS: The RDSS framework offers several groundbreaking contributions to the field of scalable verification for modular concurrent systems:

5) Verification of individual modules: The core contribution of RDSS is its ability to verify properties of individual modules using only the meta-graph of that specific module. Unlike traditional methods that require access to the global state space, RDSS is a pioneering approach that enables local model checking. This approach drastically improves scalability, as it allows the modules of large systems to be verified efficiently. The benefits of this modular verification strategy are further illustrated in Fig. 8, which compares the verification times of RDSS with those of LTSmin. The results show that RDSS achieves competitive or even superior performance, particularly in systems where modules are not all synchronized on the same transitions.

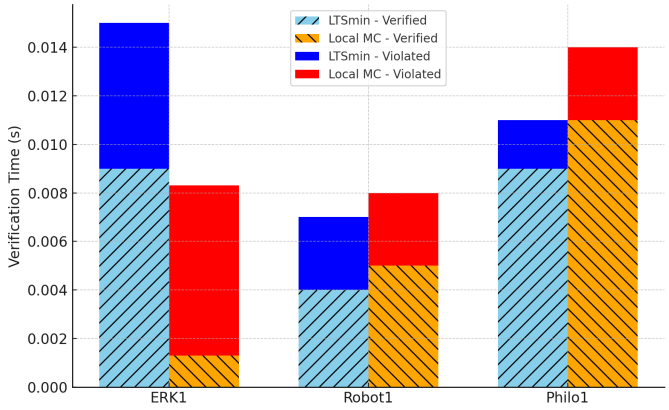


Fig. 8. Comparative performance of LTSmin and Local Model Checking in verifying modular systems.

6) Integration of state space reduction techniques: Another significant contribution of RDSS is its integration of various state space reduction techniques into a modular framework. Methods such as partial-order reduction (POR), symmetry reduction, and abstraction are applied both at the global and local levels. By combining these reductions, RDSS is a hybrid method that optimizes the verification process, ensuring the correctness of global properties while managing the complexity of individual modules.

7) Flexibility in modular verification: RDSS introduces a new level of flexibility in the verification of modular systems, particularly for systems with dynamic interactions. Unlike traditional verification methods that may struggle with complex interdependencies between modules, RDSS can efficiently handle systems where interactions evolve, leveraging its module-specific meta-graphs. Moreover, the modular structure

[2]https://github.com/chihebabid/DSS-Checker

of RDSS is well-suited for distributed concurrent systems, as it allows for efficient verification by focusing on individual modules and their interactions, rather than constructing the entire global state space.

8) State space size reduction: When compared with the flat Petri net state space, RDSS delivers better results, particularly for systems where not all modules are synchronized on the same transitions. This reduction in state space is particularly beneficial for verifying large-scale modular systems, as shown in Fig. 9, which illustrates the performance improvement of RDSS over traditional methods. The comparison between the flat Petri net and RDSS in terms of transitions and markings demonstrates the scalability advantages of the RDSS approach.

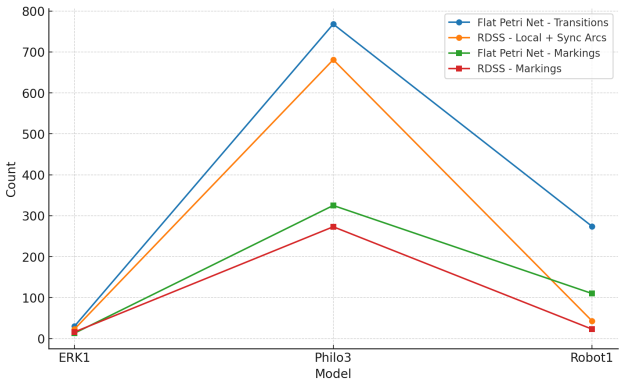


Fig. 9. Comparison of Flat Petri Net and RDSS. The figure illustrates the performance improvement of RDSS in terms of state space size reduction, particularly in systems with unsynchronized modules.

9) Model checking focus: A key strength of RDSS lies in its model checking capabilities. By focusing on module-level verification, RDSS allows for the verification of properties such as deadlock, liveness, and reachability directly within each module, using its associated meta-graph. This localized model checking significantly reduces the computational burden compared to traditional approaches that require the construction and exploration of the entire global state space.

Fig. 10 illustrates the RDSS model checking process, highlighting how the meta-graph of each module is used for verification. This diagram emphasizes the modular decomposition of the verification process, showcasing the RDSS's ability to verify individual components without needing to consider the full system's state space.

10) Worked example: To illustrate the process, consider the Modular Petri Net (MPN) in Fig. 4 and its corresponding RDSS in Fig. 7.

We aim to verify the following $LTL \setminus \mathbf{X}$ property on module A:

$$\varphi = \mathbf{G}(sync_1 \rightarrow \mathbf{F}\, t_{a1}) \ \wedge \ \mathbf{G}\big(\neg(t_{a1} \rightarrow \mathbf{F}\, t_{a2})\big).$$

Instead of constructing the global state space, verification is restricted to the meta-graph RG_A of module A (Fig. 11). The procedure follows the automata-theoretic model checking approach:

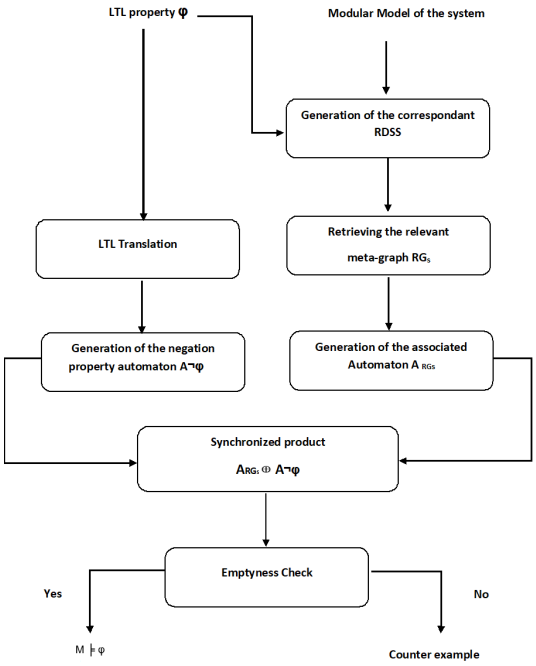


Fig. 10. The RDSS local model checking workflow.

- Negate the property φ to obtain $\neg\varphi$.
- Construct the Büchi automaton $A_{\neg\varphi}$ (Fig. 12).
- Build the automaton A_{RG_A} from the module's meta-graph.
- Compute the synchronous product $A_{RG_A} \otimes A_{\neg\varphi}$ (Fig. 13 and Fig. 14.

Initially, both automata start in their respective initial states. During the first synchronization step (Fig. 13), when the transition $sync1$ is activated in the meta-graph of module A, the automaton of $\neg\varphi$ fires $!t_{a1}$, moving to state 1. Subsequently, as illustrated in Fig. 14, the cycle in the component's strongly connected component (SCC) allows repeated firings of $t_{a2}.t_{a3}$, which are matched by the "true" loop in the automaton of $\neg\varphi$.

In this case, the product automaton is not empty. The counterexample execution is:

$$sync_1 \cdot (t_{a1}\, t_{a2})^*,$$

showing that φ is violated.

This example demonstrates concretely how RDSS enables local model checking by verifying properties using only the meta-graph of the concerned module, thus avoiding the explosion of the global state space.

These contributions, particularly the ability to perform local model checking and reduce the global state space, mark RDSS as a highly effective and scalable solution for verifying large, modular systems.

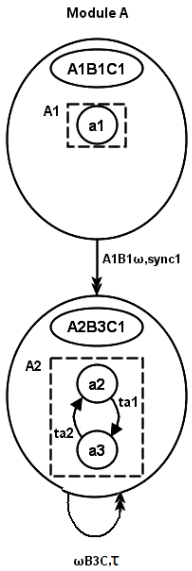


Fig. 11. Meta-graph of module A.

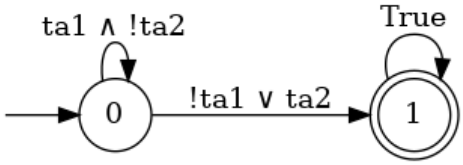


Fig. 12. Automaton of $\neg\varphi$.

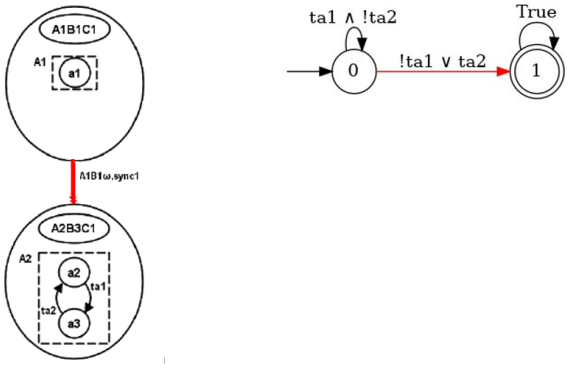


Fig. 13. First step of the synchronized product construction.

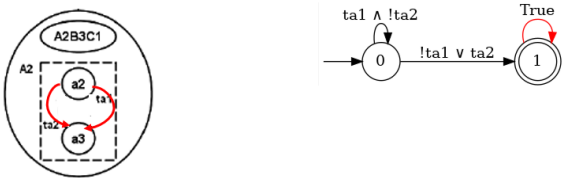


Fig. 14. Second step of the synchronized product construction.

11) Experimental setup and reproducibility: To ensure reproducibility of the reported results, we now provide additional details on the benchmarks, environment, and properties considered.

a) Benchmarks: The experiments were carried out on Modular Petri Net models representative of the case studies discussed in this paper. These include 1) an IoT-based monitoring scenario with modular communication between devices and gateways, and 2) a multi-robot patrolling system with synchronized alert transitions. For each model, different parameterizations were used to vary the number of modules and the degree of synchronization between them.

b) Verification properties: The properties evaluated include locally generic ones such as deadlock-freedom, home-space reachability, and liveness, as well as temporal properties expressed in the stutter-invariant fragment LTL$\setminus\mathbf{X}$ (e.g. $\mathbf{G}\,\neg error$ and $\mathbf{GF}\,request \rightarrow \mathbf{F}\,grant$). These properties were selected to cover both safety and liveness aspects.

c) Environment: All experiments were executed on a workstation running Ubuntu 22.04 LTS, equipped with an Intel Core i7 processor and 16 GB of RAM. The RDSS prototype was implemented in C++/ROS2, while LTSmin and SPIN were used as comparative tools. Each verification run was subject to a timeout of 30 minutes.

d) Replication: The RDSS implementation, benchmark models, and property specifications are publicly available at https://github.com/chihebabid/ros2dss, enabling replication of the results presented in this paper.

E. Positioning RDSS within the Literature

The Reduced Distributed State Space (RDSS) framework introduces a modular approach to overcome state explosion, going beyond traditional reductions by exploiting meta-graphs of individual modules. We briefly position RDSS against classical techniques and widely used tools, and highlight its application domains.

1) Reduction techniques: Compared to Partial-Order Reduction (POR), RDSS avoids redundant global interleavings by applying reductions locally at the module level. Unlike Symmetry Reduction (SR), which assumes static symmetries, RDSS flexibly combines SR with τ-hiding and meta-state fusion. In contrast to coarse-grained abstractions, RDSS preserves stuttering equivalence, ensuring correctness of LTL$\setminus X$ properties.

2) Verification tools: NuSMV is effective for small symbolic systems but does not scale to modular decomposition. LTSmin supports large state spaces yet lacks modular reduction; RDSS achieves competitive performance with lower memory needs (Fig. 8). SPIN remains powerful for explicit-state checking but does not provide modular verification like RDSS, which verifies module-specific properties directly.

3) Application domains: RDSS is particularly well-suited for distributed concurrent systems, where modules interact through synchronization but maintain partial independence. This includes multi-robot systems, IoT platforms, and cyber-physical infrastructures, where global state construction is infeasible. For example, in [82], RDSS was successfully applied to an IoT-based case study, showing that local model checking can verify safety and liveness properties efficiently without exploring the full global state space.

4) Summary: Overall, RDSS integrates the principles of POR, SR, and abstraction while overcoming their modularity limitations. By focusing on module-specific properties, RDSS positions itself as a scalable solution for distributed concurrent systems, with promising applications in robotics, IoT, and large-scale modular infrastructures. As highlighted in Table V, RDSS complements existing techniques and tools by enabling local model checking with reduced state space construction, offering clear advantages in modular and dynamic settings.

F. Limits Under Dense Synchronization

While RDSS demonstrates clear advantages when modules retain partial independence, a pathological case arises when the system exhibits dense synchronization, i.e. when most or all transitions are shared among modules (case of ERK1). In this scenario, the benefits of modular decomposition diminish, since local meta-graphs must frequently synchronize, reducing the reduction opportunities from τ-hiding and meta-state fusion.

Compared to classical techniques, RDSS in this setting behaves closer to a flat global exploration, as synchronization dominates the structure of the meta-graphs. Partial-order reduction (POR) or stubborn sets may then achieve comparable or even better pruning, since they exploit independence at the transition level rather than structural modularity. Symmetry reduction may also remain beneficial if replicated modules synchronize in a uniform manner, collapsing symmetric synchronization patterns.

These observations suggest that RDSS is most advantageous in systems where:

- Modules contain a significant proportion of local transitions relative to their synchronized ones,
- Synchronization points are sparse or limited to subsets of modules, and
- Property checking targets module-specific behaviors (e.g. local deadlock, local liveness, LTL$\setminus X$ properties tied to one module).

In contrast, in systems with dense synchronization, hybrid approaches that combine RDSS with stubborn sets or symbolic encodings may be more appropriate, ensuring that both structural and transition-level redundancies are exploited.

G. Future Directions and Extensions

While the RDSS framework already provides a scalable and modular verification approach, several promising research directions remain open. These extensions aim to further enhance scalability, applicability, and integration with modern verification practices.

1) Distributed implementations with ROS2: One natural extension of RDSS is its deployment in distributed verification environments. A first step in this direction has been demonstrated through a prototype implementation based on ROS2, where modules act as independent nodes communicating through publish/subscribe services. This distributed construction of the RDSS allows verification tasks to be executed concurrently across multiple workers, offering increased scalability for large systems with heterogeneous components.[3] Future work can explore optimizations in communication overhead and dynamic load balancing.

2) AI-driven heuristics for reduction: The construction of RDSS relies on reduction techniques, such as partial-order reduction, symmetry, and abstraction. Integrating artificial intelligence, particularly reinforcement learning and machine learning heuristics, could guide the selection of reduction strategies dynamically. For instance, AI agents may learn to prioritize transitions, identify symmetries, or detect redundant synchronizations, thereby reducing exploration costs while preserving correctness. Such integration would pave the way for adaptive and context-aware verification pipelines.

3) Hybrid reduction strategies: Another promising direction is the design of hybrid approaches that combine RDSS with other established reduction frameworks. This includes integrating counterexample-guided abstraction refinement (CEGAR), advanced stubborn sets, or symbolic encodings with the modular meta-graph construction of RDSS. Such hybrids would allow leveraging the strengths of different techniques to achieve stronger reductions without sacrificing property preservation, particularly for LTL$\setminus X$ and branching-time logics.

4) Application to emerging domains: RDSS is naturally suited for concurrent and modular architectures such as multi-robot coordination, IoT ecosystems, and cloud-native distributed systems. Extending RDSS to these domains can provide realistic benchmarks that highlight its advantages over state-of-the-art tools. For example, in the IoT-based agriculture use case [82], RDSS enabled local model checking without requiring the full global state. Similar extensions to autonomous vehicles, cyber-physical systems, and large-scale cloud infrastructures would consolidate RDSS as a practical solution to real-world scalability challenges.

5) From local to global verification: While RDSS focuses primarily on enabling local model checking at the module level, an important extension is to move toward global verification within the same modular framework. The key idea is that a global property can often be decomposed into a set of local properties, each verifiable on the corresponding module's meta-graph. By invoking local model checking for each component property and then combining the results, one can infer the truth of the global property.

This modular synthesis of verification outcomes effectively reconstructs global correctness from local evidence, avoiding the need to explore the full global state space. Such an approach bridges local and global reasoning, and can be viewed as a form of modular controller synthesis, where global guarantees emerge from the coordinated satisfaction of local obligations.

The verification of modular concurrent systems is no longer an abstract academic pursuit but a practical necessity for the safe deployment of autonomous platforms, IoT infrastructures, and large-scale distributed software. Ensuring correctness at scale requires techniques that tame the state explosion problem without sacrificing soundness.

[3]Prototype implementation available at: https://github.com/chihebabid/ros2dss

TABLE V. POSITIONING RDSS WITH RESPECT TO CLASSICAL TECHNIQUES AND TOOLS

Approach / Tool	Main Principle	Limitations	RDSS Perspective
Partial-Order Reduction (POR)	Prunes redundant interleavings by exploiting independence of concurrent transitions	Less effective with synchronous transitions across modules	Applied locally at module level to avoid global blow-up
Symmetry Reduction (SR)	Collapses symmetric states from replicated components	Assumes static symmetry; weak for dynamic or partial symmetries	Combined with τ-hiding and meta-state fusion to preserve behaviors
Abstraction	Groups concrete states into abstract representatives (predicate/counter abstraction)	Risk of oversimplification; may lose critical distinctions	Balanced with stuttering equivalence, ensuring correctness of LTL$\setminus X$
NuSMV	Symbolic model checking with BDD/SAT	Limited scalability for modular decomposition	RDSS avoids global symbolic encoding, scales better for Modular Petri Nets
LTSmin	Symbolic state-space exploration	Handles large systems, but lacks modular reductions	RDSS achieves competitive verification times with lower memory usage (see Fig. 8)
SPIN	Explicit-state temporal logic model checking	No modular decomposition; explores entire global space	RDSS verifies module-specific properties directly using local meta-graphs

In this survey, we have examined the landscape of techniques for scalable verification, including state space reduction, abstraction, compositional reasoning, symbolic methods, and distributed verification. Within this context, we have introduced and positioned the Reduced Distributed State Space (RDSS) framework as a novel and effective contribution. RDSS reduces the global state space into modular meta-graphs, enforces stuttering equivalence, and supports verification at the module level. This enables properties to be checked locally, a capability not offered by existing approaches. Our experimental comparisons against powerful tools such as LTSmin demonstrate the scalability gains of RDSS, particularly when modules are not all synchronized on the same transitions. Furthermore, we have shown that RDSS integrates naturally with techniques such as partial-order reduction, symmetry reduction, and abstraction, achieving strong reductions while maintaining correctness.

Beyond these contributions, the RDSS framework has been extended through a distributed prototype built on ROS2, opening the way to practical deployment in real-world modular systems. Its modular structure makes it especially suitable for verifying concurrent and distributed systems, ranging from multi-robot applications to cloud-based infrastructures.

While challenges such as state explosion and synchronization overhead remain, the trajectory of research indicates a strong movement toward hybrid approaches, combining symbolic encodings, distributed infrastructures, and reduction-based methods. The synergy between theoretical foundations and real-world applications promises a future where scalable verification becomes a standard component of the engineering of next-generation systems. RDSS contributes to this vision by providing a modular, scalable, and extensible verification framework that addresses both academic challenges and industrial needs.

VII. THREATS TO VALIDITY

As with any empirical and methodological contribution, our evaluation and claims are subject to certain threats to validity. We outline the most relevant ones below.

A. Model Representativeness

The benchmarks used (e.g. ERK1, Philo3, Robot1, IoT case studies) are drawn from widely used academic examples of modular and concurrent systems. However, they may not capture the full diversity of industrial-scale systems, especially those with heterogeneous architectures or non-Petri net formalisms. Future work should incorporate larger and more varied case studies to ensure broader coverage.

B. Property Selection Bias

Our focus on stuttering-invariant $LTL \setminus X$ properties, such as deadlock-freedom, local liveness, and reachability, reflects the classes of properties naturally supported by RDSS. Nevertheless, this may bias the evaluation, as other properties (e.g. CTL, fairness, or real-time constraints) are not currently addressed. Extending RDSS to these classes remains an important direction.

C. Tool Configuration Bias

Comparisons with state-of-the-art tools such as LTSmin or SPIN depend on configuration choices (e.g. symbolic encodings, partial-order options, memory allocation). While we sought fair defaults, alternative configurations might yield different relative performance. A more exhaustive benchmarking campaign would provide stronger evidence.

D. Generality Across Formalisms

RDSS is defined over Modular Petri Nets (MPNs), which provide a natural and expressive formalism for concurrent systems. This focus raises a validity threat regarding generality: it remains to be shown to what extent RDSS principles carry over to other formalisms, such as process algebras, timed automata, or hardware verification languages. Initial indications are promising, but further formalization is required.

Overall, these threats highlight the need for additional validation, both empirically (through more benchmarks and diverse case studies) and theoretically (by extending RDSS to broader classes of systems and properties). We report them here to provide a transparent account of the current scope and limitations of our contribution.

VIII. CONCLUSION

The verification of modular concurrent systems is no longer an abstract academic pursuit but a practical necessity for the safe deployment of autonomous platforms, IoT infrastructures, and large-scale distributed software. Ensuring correctness at

scale requires techniques that tame the state explosion problem without sacrificing soundness.

In this survey, we have examined the landscape of techniques for scalable verification, including state space reduction, abstraction, compositional reasoning, symbolic methods, and distributed verification. Within this context, we have introduced and positioned the Reduced Distributed State Space (RDSS) framework as a novel and effective contribution. RDSS reduces the global state space into modular meta-graphs, enforces stuttering equivalence, and supports verification at the module level. This enables properties to be checked locally, a capability not offered by existing approaches. Our experimental comparisons against powerful tools such as LTSmin demonstrate the scalability gains of RDSS, particularly when modules are not all synchronized on the same transitions. Furthermore, we have shown that RDSS integrates naturally with techniques such as partial-order reduction, symmetry reduction, and abstraction, achieving strong reductions while maintaining correctness.

Beyond these contributions, the RDSS framework has been extended through a distributed prototype built on ROS2, opening the way to practical deployment in real-world modular systems. Its modular structure makes it especially suitable for verifying concurrent and distributed systems, ranging from multi-robot applications to cloud-based infrastructures.

Nevertheless, several challenges remain open. Scalability limits may reappear when dealing with extremely large or highly interconnected modular systems, where synchronization overhead can dampen the benefits of reduction. Moreover, while RDSS preserves stuttering equivalence, there are trade-offs between accuracy and efficiency, particularly when aggressive fusion of meta-states or τ-hiding strategies are applied. Another promising direction lies in complementing RDSS with AI-driven heuristics that guide exploration or with hybrid reductions that combine symbolic encodings and distributed infrastructures. Such extensions would strengthen the adaptability of RDSS across diverse system classes and push its applicability to industrial-scale deployments.

The synergy between theoretical foundations and real-world applications promises a future where scalable verification becomes a standard component of the engineering of next-generation systems. RDSS contributes to this vision by providing a modular, scalable, and extensible verification framework that addresses both academic challenges and industrial needs, while opening avenues for innovation in hybrid and AI-augmented verification methods.

REFERENCES

[1] A. Valmari, "The state explosion problem," in W. Reisig and G. Rozenberg, Eds., *Lectures on Petri Nets I: Basic Models (ACPN 1996), Lecture Notes in Computer Science*, vol. 1491. Berlin, Heidelberg: Springer, 1998, pp. 429–528, doi: 10.1007/3-540-65306-6_21.

[2] S. Alagar and S. Venkatesan, "Techniques to tackle state explosion in global predicate detection," *IEEE Transactions on Software Engineering*, vol. 27, no. 8, pp. 704–714, Aug. 2001, doi: 10.1109/32.940566.

[3] S. C. Sølvsten and J. van de Pol, "Symbolic Model Checking in External Memory," preprint, May 2025.

[4] I. Marmanis, M. Kokologiannakis, and V. Vafeiadis, "Model Checking C/C++ with Mixed-Size Accesses," *Proc. ACM Program. Lang.*, vol. 9, no. POPL, Art. 75, Jan. 2025, 21 pp., doi: 10.1145/3704911.

[5] M. Dam and H. Nemati, "Compositional Verification of Concurrency Using Past-Time Temporal Epistemic Logic," preprint, Feb. 2025.

[6] J. Srba, "On-The-Fly Verification: Advancements in Dependency Graphs (Invited Talk)," in *Proc. 36th International Conference on Concurrency Theory (CONCUR 2025)*, Leibniz International Proceedings in Informatics (LIPIcs), vol. 348, pp. 3:1–3:5, Dagstuhl, Germany, 2025, doi: 10.4230/LIPIcs.CONCUR.2025.3.

[7] G. Kant, A. Laarman, J. Meijer, J. van de Pol, S. Blom, and T. van Dijk, "LTSmin: High-Performance Language-Independent Model Checking," in *Proc. 21st International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, Lecture Notes in Computer Science, vol. 9035, pp. 692–707, Springer, 2015, doi: 10.1007/978-3-662-46681-0_61.
in *Proc. Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, Lecture Notes in Computer Science, vol. 9035, 2015. Extended overview of recent (2024) multi-core symbolic and partial-order reduction back-ends.

[8] G. J. Holzmann, *The SPIN Model Checker: Primer and Reference Manual*. Boston, MA: Addison–Wesley, 2004.

[9] C. A. Abid, K. Klai, H. Ouni, and M. K. Kholladi, "Distributed Modular Verification Based on Symbolic Observation Graphs," *International Journal of Critical Computer-Based Systems*, vol. 10, no. 1, pp. 36–60, 2020, doi: 10.1504/IJCCBS.2020.110498.

[10] A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, and A. Tacchella, "NuSMV 2: An OpenSource Tool for Symbolic Model Checking," in *Proc. 14th Int. Conf. on Computer Aided Verification (CAV)*, Lecture Notes in Computer Science, vol. 2404, pp. 359–364, Springer, 2002, doi: 10.1007/3-540-45657-0_29.

[11] E. M. Clarke, O. Grumberg, and D. A. Peled, *Model Checking*. Cambridge, MA, USA: MIT Press, 1999.

[12] C. Baier and J.-P. Katoen, *Principles of Model Checking*. Cambridge, MA, USA: MIT Press, 2008.

[13] S. Christensen and L. Petrucci, "Modular state space analysis of coloured Petri nets," in *Lectures on Petri Nets I: Basic Models (ACPN 1996), Lecture Notes in Computer Science*, vol. 1491. Berlin, Heidelberg: Springer, 1998, pp. 429–528, doi: 10.1007/3-540-65306-6_21.

[14] C.Edmund and Emerson, E. (2008). "Design and Synthesis of Synchronization Skeletons Using Branching Time Temporal Logic", in Lecture Notes Comput Sci. 131. 196-215.

[15] Andrés, Miguel. Quantitative Analysis of Information Leakage in Probabilistic and Nondeterministic Systems. Diss. Radboud University, Nijmegen, 2011.

[16] E. M. Clarke, P. Huang, M. Nováček, M. Paulk, B. Schuppan, and P. Zuliani, "Model Checking and the State Explosion Problem," Tech. Rep., Carnegie Mellon University, 2011.

[17] Li, Shuo, et al. "On-the-fly unfolding with optimal exploration for linear temporal logic model checking of concurrent software and systems." Automated Software Engineering 32.2 (2025): 60.

[18] F. Herbreteau, S. Larroze-Jardiné, G. Point, and I. Walukiewicz, "Revisiting Stateful Partial-Order Reduction," 2024.

[19] R. Cleaveland and C. Trippel, "Memory Consistency Model-Aware Cache Coherence for Heterogeneous Hardware," in *Proc. 25th Int. Conf. on Formal Methods in Computer-Aided Design (FMCAD)*, 2024.

[20] S. Merz, "Stuttering Equivalence and Stuttering Invariance," Inria Nancy & LORIA Technical Report, March 17, 2025.

[21] T. R. Nicely, "Pentium FDIV flaw," Tech. Rep., Lynchburg College, 1995.

[22] N. G. Leveson and C. S. Turner, "An Investigation of the Therac-25 Accidents," *Computer*, vol. 26, no. 7, pp. 18–41, IEEE, 1993, doi: 10.1109/MC.1993.274940.

[23] House Committee on Transportation and Infrastructure, "The Boeing 737 MAX Aircraft: Costs, Consequences, and Lessons from Its Design, Development, and Certification," U.S. Government Report, 2020.

[24] S. Appicharla, "Analyses of the Boeing 737 MAX accidents: formal models and safety implications," *Sociotechnical Systems Journal*, vol. 12, pp. 1–15, 2023.

[25] National Transportation Safety Board (NTSB), "Collision Between Vehicle Controlled by Developmental Automated Driving System and Pedestrian," Highway Accident Report NTSB/HAR-19/03, 2019.

[26] P. Kocher et al., "Spectre Attacks: Exploiting Speculative Execution," in *Proc. 40th IEEE Symposium on Security and Privacy (S&P)*, pp. 1–19, 2019, doi: 10.1109/SP.2019.00002.

[27] Nasrabadi, Faezeh, Robert Kuennemann, and Hamed Nemati. "Symbolic Parallel Composition for Multi-language Protocol Verification." 2025 IEEE 38th Computer Security Foundations Symposium (CSF). IEEE Computer Society, 2025.

[28] Amazon Web Services, "Summary of the AWS Service Event in the Northern Virginia (US-EAST-1) Region," Incident Report, Dec. 2021.

[29] C. Kalaskar and T. S., "Fault tolerance of cloud infrastructure with machine learning," *Cybernetics and Information Technologies*, vol. 23, no. 4, pp. 90–103, 2023, doi:10.2478/cait-2023-0034.

[30] Author Unknown, "Optimizing Resource Usage in Cloud Infrastructure," *Proc. ACM Symposium on Cloud Computing*, June 2025.

[31] C. Coti, L. Petrucci, C. Rodríguez, M. Sousa, and H. T. T. Nguyen, "Quasi-Optimal Partial Order Reduction," *Formal Methods in System Design*, vol. 57, pp. 3–33, 2021, doi: 10.1007/s10703-020-00350-4.

[32] E. Albert, L. Gómez, and A. Legay, "Optimal Context-Sensitive Dynamic Partial-Order Reduction with Observers," *Journal of Systems Architecture*, vol. 144, 2023, doi: 10.1016/j.sysarc.2023.102853.

[33] X. Yang and C. Ye, "Dynamic Partial Order Reduction Based on Interference Solution," *Research Square*, preprint, Version 1, 18 Sept. 2023. doi: 10.21203/rs.3.rs-3257081/v1.

[34] T. Gibson-Robinson and G. Lowe, "Symmetry Reduction in CSP Model Checking," *Int. Journal on Software Tools for Technology Transfer*, vol. 21, no. 4, pp. 481–496, 2019, doi:10.1007/s10009-019-00516-4.

[35] R. Barbosa, A. Fonseca, and F. Araujo, "Reductions and Abstractions for Formal Verification of Distributed Round-Based Algorithms," *Software Quality Journal*, vol. 29, no. 3, pp. 705–731, 2021, doi:10.1007/s11219-020-09539-6.

[36] P. Eichler, S. Jacobs, and C. Weil-Kennedy, "Parameterized Verification of Systems with Precise (0,1)-Counter Abstraction," in *Proc. 26th Int. Conf. on Verification, Model Checking, and Abstract Interpretation (VMCAI)*, LNCS, vol. 15529, pp. 101–124, Springer, 2025.

[37] T. Neele, A. Valmari, and A. Wijs, "Stubborn Set Reduction for Two-Player Reachability Games," in *Proc. 2021 Int. Conf. on Application of Concurrency to System Design (ACSD)*, pp. 23–33, IEEE, 2021. doi: 10.1109/ACSD51164.2021.00011.

[38] S. Ben Rhouma, K. Klai, and H. Ben-Abdallah, "On Improving Model Checking of Time Petri Nets and Its Application to the Formal Verification," *Int. J. of Information System Modeling and Design (IJISMD)*, vol. 12, no. 3, pp. 1–22, 2021. doi: 10.4018/IJISMD.20210701.oa1.

[39] Y. Kim, W. Jamroga, W. Penczek, and L. Petrucci, "Practical Abstractions for Model Checking Continuous-Time Multi-Agent Systems," in *Proc. AAMAS 2025*, May 2025.

[40] G. Bruns, "User-Driven Abstraction for Model Checking," preprint, 2023. . Available: https://arxiv.org/abs/2307.15820

[41] A. Benveniste et al., "Contracts for System Design," *Found. Trends Electron. Des. Autom.*, vol. 12, no. 2-3, pp. 124–400, 2018. doi: 10.1561/1000000053.

[42] K. Pierce et al., "Automated Contract Composition and Refinement for System-on-Chip Integration," in *Proc. Design, Autom. Test Eur. Conf. (DATE)*, 2023, pp. 1–6. doi: 10.23919/DATE56975.2023.10137125.

[43] N. Tabassam et al., "Contract-Based Design Methodologies for Cyber-Physical Systems," in *Proc. Int. Conf. World Conf. Smart Trends Syst., Secur. Sustainability (WorldS4)*, Springer, Singapore, 2025.

[44] O. Goudsmid, O. Grumberg, and S. Sheinvald, "Compositional Model Checking for Multi-Properties," in *Proc. Int. Conf. Verif., Model Checking, Abstract Interpret. (VMCAI)*, 2021, pp. 1–23. doi: 10.1007/978-3-030-67067-2_1

[45] R. Krauss, M. Goli, and R. Drechsler, "EDDY: A Multi-Core BDD Package with Dynamic Memory Management and Reduced Fragmentation," in *Proc. 28th Asia South Pac. Des. Autom. Conf. (ASP-DAC)*, 2023, pp. 1–6.

[46] O. S. Bak et al., "GPU Accelerating Statistical Model Checking for Extended Timed Automata," in *Principles of Verification: Cycling the Probabilistic Landscape: Essays Dedicated to Joost-Pieter Katoen on the Occasion of His 60th Birthday, Part II*, Cham: Springer Nature Switzerland, 2024, pp. 267–292.

[47] S. Evangelista, L. M. Kristensen, and L. Petrucci, "Evaluation of a Distributed Explicit State Space Exploration Algorithm with State Reconstruction for RDMA Networks," *Int. J. Softw. Tools Technol. Transfer*, vol. 27, no. 2, pp. 149–168, 2025.

[48] L. Davis and T. Ji, "Evaluating SAT and SMT Solvers on Large-Scale Sudoku Puzzles,"preprint 2025.

[49] Spork, Timm, et al. "A Spectrum of Approximate Probabilistic Bisimulations." 35th International Conference on Concurrency Theory. 2024.

[50] G. Behrmann, A. David, and K. G. Larsen, "A Tutorial on UPPAAL," *Formal Methods in System Design*, vol. 23, no. 1, pp. 47–104, 2003, doi: 10.1023/A:1023688603802.

[51] I. Grobelna, K. Gajewski, and A. Karatkevich, "A Systematic Review on the Applications of UPPAAL," *Sensors*, vol. 25, no. 11, p. 3484, 2025, doi: 10.3390/s25113484.

[52] J. Laarman, D. van de Pol, and J. Wijs, "LTSmin: High-Performance Language-Independent Model Checking," in *Proc. 23rd Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, LNCS, vol. 10205, pp. 692–707, Springer, 2017, doi: 10.1007/978-3-662-54577-5_9.

[53] J. Barnat, L. Brim, and P. Ročkai, "DiVinE: Parallel Distributed Model Checker," in *Proc. Int. Conf. on High Performance Computing (HiPC)*, pp. 11–20, IEEE, 2010, doi: 10.1109/HIPC.2010.5696880.

[54] C. Hensel, J.-P. Katoen, J. Krčál, F. Olmedo, J. Wojtczak, and L. Zhang, "The STORM Model Checker: A Modern, Symbolic, and Flexible Framework for Probabilistic Verification," *International Journal on Software Tools for Technology Transfer (STTT)*, vol. 24, no. 2, pp. 205–227, 2022, doi: 10.1007/s10009-021-00633-z.

[55] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of Probabilistic Real-Time Systems," in *Proc. 23rd Int. Conf. on Computer Aided Verification (CAV)*, LNCS, vol. 6806, pp. 585–591, Springer, 2011, doi: 10.1007/978-3-642-22110-1_47.

[56] J. Sun, Y. Liu, J. S. Dong, and J. Pang, "PAT: Towards Flexible Verification under Fairness," in *Proc. 21st Int. Conf. on Computer Aided Verification (CAV)*, LNCS, vol. 5643, pp. 709–714, Springer, 2009, doi: 10.1007/978-3-642-02658-4_50.

[57] T. Ball, R. Majumdar, T. Millstein, and S. K. Rajamani, "Automatic Predicate Abstraction of C Programs," *ACM SIGPLAN Notices*, vol. 47, no. 4a, pp. 37–47, 2012, doi: 10.1145/2103621.2103627.

[58] M. Kokologiannakis, I. Marmanis, and V. Vafeiadis, "SPORE: combining symmetry and partial order reduction," *Proc. ACM on Programming Languages*, vol. 8, no. PLDI, pp. 1781–1803, 2024,

[59] M. Rashid, A. Ali, S. A. Shah, and F. Ahmed, "Formal modeling and verification of IoT-based smart transport system using SPIN model checker," in *Proc. 2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC)*, Karachi, Pakistan, pp. 1–6, IEEE, 2024, doi: 10.1109/KHI-HTC61644.2024.10722564.

[60] N. Suresh Kumar and G. Santhosh Kumar, "Abstracting IoT protocols using timed process algebra and SPIN model checker," *Cluster Computing*, vol. 26, no. 2, pp. 1611–1629, 2023, doi: 10.1007/s10586-022-03769-5.

[61] Y. Yang and T. Holvoet, "Efficient Model Checking with Semantically-Equivalent Models for vGOAL," in *Proc. 24th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1–9, IFAAMAS/ACM, 2025.

[62] S. Kanwal and W. Fokkink, "A Framework for Model-Based Specification and Verification in Feature-Oriented Software Product Lines," in *Proc. Int. Conf. on Fundamentals of Software Engineering (FSEN)*, Cham, Switzerland: Springer Nature, 2025, pp. 1–18.

[63] M. Krichen, "Timed automata-based strategy for controlling drone access to critical zones: A UPPAAL modeling approach," *Electronics*, vol. 13, no. 13, p. 2609, 2024, doi: 10.3390/electronics13132609.

[64] D. Basile, "Modelling, verifying and testing the contract automata runtime environment with UPPAAL," in *Proc. Int. Conf. on Coordination Models and Languages (COORDINATION)*, Cham, Switzerland: Springer Nature, 2024, pp. 123–139.

[65] M. Osama and A. Wijs, "Hitching a ride to a lasso: massively parallel on-the-fly LTL model checking," in *Proc. Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, Cham, Switzerland: Springer Nature, 2024, pp. 145–165.

[66] A. Duret-Lutz, D. Poitrenaud, and Y. Thierry-Mieg, "Simplifying LTL Model Checking Given Prior Knowledge," in *Proc. Int. Conf. on Applications and Theory of Petri Nets and Concurrency (PETRI NETS)*, Cham, Switzerland: Springer Nature, 2025, pp. 45–65.

[67] F. R. Monteiro, M. R. Gadelha, and L. C. Cordeiro, "Model checking C++ programs," *Software Testing, Verification and Reliability*, vol. 32, no. 1, e1793, 2022.

[68] R. van Spreuwel and A. Wijs, "No need to be stubborn: partial-order reduction for GPU model checking revisited," in *Proc. Int. Symp. on Leveraging Applications of Formal Methods (ISoLA)*, Cham, Switzerland: Springer Nature, 2024, pp. 233–251.

[69] Y. Gao, D. Xu, and L. Chen, "Prediction and optimization of memory fragmentation based on eBPF and LightGBM in high-load environments," in *Proc. 2024 7th Int. Conf. on Artificial Intelligence and Pattern Recognition (AIPR)*, pp. 342–350, ACM, 2024. doi: 10.1145/3670810.3670924.

[70] Z. Huang, Y. Zhou, L. Wang, and J. Zhang, "Reducing GPU Memory Fragmentation via Spatio-Temporal Planning for Efficient Large-Scale Model Training," 2025 . Available: https://arxiv.org/abs/2507.16274

[71] M. Lampacrescia, M. Klauck, and M. Palmas, "Towards verifying robotic systems using statistical model checking in STORM," in *Proc. Int. Conf. on Bridging the Gap between AI and Reality (AIReality)*, Cham, Switzerland: Springer Nature, 2024, pp. 89–106.

[72] T. Quatmann, "What is the best algorithm for MDP model checking?," *International Journal on Software Tools for Technology Transfer (STTT)*, pp. 1–7, 2025, doi: 10.1007/s10009-025-00791-3.

[73] I. Valkov, A. F. Donaldson, and A. Miller, "Synchronisation in Language-Level Symmetry Reduction for Probabilistic Model Checking," in *Proc. Int. Symp. on Model Checking Software (SPIN)*, Cham, Switzerland: Springer Nature, 2024, pp. 210–228.

[74] H. Wang, Y. Zhao, X. Liu, and L. Zhang, "Towards a Model-Based Framework for Automated Traceable Systems and Probabilistic Model Checking," in *Proc. 2025 6th Int. Conf. on Computer Engineering and Application (ICCEA)*, IEEE, pp. 56–63, 2025. doi: 10.1109/ICCEA61623.2025.10789765.

[75] M. Djukic, D. Stojanovic, J. Protic, and Z. Rakic, "Correct orchestration of federated learning generic algorithms: Python translation to CSP and verification by PAT," *International Journal on Software Tools for Technology Transfer*, vol. 27, no. 1, pp. 21–34, 2025, doi: 10.1007/s10009-024-00789-5.

[76] Y. Yu, X. Zhou, Z. Liu, and Z. Yang, "Model checking concurrency in smart contracts with a case study of safe remote purchase," in *Proc. Int. Conf. on Formal Engineering Methods (ICFEM)*, Singapore: Springer Nature, 2024, pp. 145–161.

[77] P. Chatterjee, B. Jobstmann, D. Kroening, and F. Somenzi, "Distributed bounded model checking," *Formal Methods in System Design*, vol. 64, no. 1, pp. 50–72, 2024, doi: 10.1007/s10703-023-00421-9.

[78] R. Queiroz, A. Author2, A. Author3, and A. Author4, "A driver-vehicle model for ADS scenario-based testing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 8641–8654, 2024, doi: 10.1109/TITS.2023.3341234.

[79] S. Khlifa, C. A. Abid, and B. Zouari, "Local Model Checking on a Modular System," in *Proc. 10th Int. Conf. on Control, Decision and Information Technologies (CoDIT)*, IEEE, 2024, pp. 2899–2904. doi: 10.1109/CoDIT60129.2024.10563056.

[80] H. Ouni, C. A. Abid, and B. Zouari, "A Distributed State Space for Modular Petri Nets," in *Proc. 7th Int. Conf. on Modelling, Identification and Control (ICMIC)*, IEEE, 2015, pp. 1–6. doi: 10.1109/ICMIC.2015.7409426.

[81] S. Khlifa, C. A. Abid, and B. Zouari, "A Reduced Distributed State Space for Modular Petri Nets," in *Proc. Int. Conf. on Advanced Information Networking and Applications (AINA)*, Cham, Switzerland: Springer, 2023, pp. 319–331. doi: 10.1007/978-3-031-27077-2_28.

[82] S. Khlifa, C. A. Abid, A. B. Letaifa, and B. Zouari, "Local Model Checking on an IoT-Based System: Use Case of Cellular M2M in Agriculture," in *Proc. Int. Conf. on Advanced Information Networking and Applications (AINA)*, Cham, Switzerland: Springer, 2025, pp. 356–367. doi: 10.1007/978-3-031-59656-8_28.

New Explainable Overlapping Co-Clustering for Recommender Systems: Capturing Multifaceted Preferences with Enhanced Interpretability

Chiheb Eddine Ben Ncir, Mohammed Ibrahim Alattas
College of Business, University of Jeddah, Saudi Arabia

***Abstract*—Recommender systems have become critical tools in reducing information overload by providing personalized recommendations across several application domains including commerce, industry, education, academic research, etc. Clustering-based recommender systems, which use the clustering technique to group similar users or items to generate suggestions, have shown high accuracy and efficiency. However, conventional clustering methods often fail to address several challenges such as ignoring the possibility that a user may have different item preferences, limited interpretability of generated suggestions, and the inability to tailor recommendation list sizes to individual user needs. To address all these issues, we propose in this work a new recommender system based on Overlapping Co-clustering and Modularity Maximisation (OCCMM). The proposed method allows to take into account that users may have several item preferences by building overlapping clusters rather than the conventional non-overlapping model. Also, the proposed method adopts a simultaneous clustering of items and users to facilitate the generation and interpretation of suggestions through using the co-clustering technique. Furthermore, OCCMM enables an adjustment of recommendation list sizes based on an easy tunning parameter δ. Experiments conducted in three real-world datasets demonstrated the effectiveness of OCCMM in achieving better performance in terms of accuracy and interpretability compared to conventional existing methods.**

***Keywords*—Clustering-based recommender systems; modularity maximization; overlapping co-clustering; multiple-user-preferences; recommendation interpretability**

I. Introduction

Recommender systems play a vital role in addressing the challenge of information overload and providing personalized recommendations in various domains. These systems have become valuable tools for companies to deliver targeted information to users or customers. The implementation of a high-performance recommender system is usually based on various machine-learning methods and techniques such as clustering which allows for the grouping of similar users or items into the same cluster. The generation of recommendations is based on similar items or users belonging to the same cluster. Such Clustering-based recommender systems are usually characterized by an improved running time with high accuracy of recommendations [1].

Typically, clustering methods used in recommender systems have focused on either user-based or item-based recommendations. User-based recommendations focus on similarities between users where recommendations are generated based on the preferences and behavior of similar users. This recommendation approach is relatively simple to implement as it only requires user-item interaction data. However, it faces challenges when dealing with new users who have limited data or when recommending items with limited user feedback, which leads to less accurate recommendations. On the other side, item-based recommendations focus on similar items rather than users and tend to be more stable as item preferences are less likely to change compared to user preferences. Item-based recommendations allow to effectively handle new items since similar item characteristics can be used to recommend them. However, this approach may lead to less personalized recommendations since it focuses on item-item similarities and ignores user preferences.

To take advantage of both user-based and item-based approaches, the co-clustering technique can be used rather than using the clustering technique. The Co-clustering allows for simultaneous grouping of users and items allowing recommendations to be based on both characteristics simultaneously [2] [3]. However, most existing co-clustering methods assume non-overlapping co-clusters and do not allow overlaps between groups. Nevertheless, in real-life applications, users may have multiple preferences and items can satisfy multiple user needs resulting in overlapping co-clusters. Overlapping co-clustering can yield more accurate recommendation results by capturing all user's preferences and item needs. Additionally, a shortcoming with existing clustering-based recommendation methods is the absence of options for regulating the size of the recommendation list. This lack of control hampers the flexibility of the recommendation system, making it challenging to customize the number of recommendations according to user preferences.

To address all these challenges, this paper proposes a dual contribution at both the theoretical and practical levels. From a theoretical perspective, it advances the literature on recommender systems by introducing a novel formulation of the Overlapping Co-Clustering with Modular Maximization (OCCMM) framework. Unlike existing works, the proposed model integrates 1) an enhanced algorithmic formulation that optimizes modularity while supporting overlapping user–item clusters, 2) an explicit interpretability mechanism that allows practitioners to understand why certain recommendations are generated, and 3) a tunable list sizing strategy that adapts dynamically to users' contextual needs. From a practical perspective, the framework proposes explainable recommendations in domains where transparency is critical, such as e-commerce, tourism, and educational platforms. We will show the ability of the proposed OCCMM model not only to generate explanations that are transferable and generalizable to other application areas.

The rest of the paper is organized as follows: Section II presents the fundamental concepts of recommender systems. Section III discusses the challenges addressed by the proposed approach. Section IV provides a detailed theoretical framework for the overlapping co-clustering method based on modularity maximization. Section V presents the experimental results and evaluates the performance of our approach using real-world datasets. Finally, Section VI concludes the paper and outlines directions for future research.

II. Fundamental Concepts of Recommender Systems

With the exponential growth of available information across databases, web pages, and social networks, the task of extracting valuable information for users has become increasingly challenging. Users often struggle to quickly and easily find items of interest in this huge amount of available data. To address this issue, automatic recommendation tools, known as Recommender Systems (RS), have been developed. The primary objective of these systems is to minimize the time of searching and providing personalized suggestions. By analyzing extensive amounts of data through a variety of techniques and algorithms [4], these computing-based systems try to generate personalized recommendations for products or services based on individual user preferences [5]. These recommendations can span a wide range of categories, such as articles to read, products to purchase, music to listen to, movies to watch, or web pages to visit.

The effectiveness of recommendations and suggestions mainly rely on the system input data types which can be categorized into two main types: Explicit and Implicit Feedback Data. Explicit feedback data refers to information provided by users in a clear and detailed manner, typically in the form of ratings given to a set of products. This type of data directly expresses the user's preferences. On the other hand, implicit feedback data is automatically collected based on user behavior and then transformed into user preferences. Examples of implicit feedback data include clicks on links, browsing history, the number of times a song is played, or the percentage of a web page scrolled. These data are collected, converted into user preferences, and then utilized to generate recommendations.

The collected data are often represented on a scale to indicate the level of interest in a particular item. It can take various forms, such as continuous values, as seen in the Jester joke recommendation engine [6], where jokes are rated on a scale from -10 to 10. Data can also be modeled using intervals, where the lowest value represents a strong dislike and the highest indicates a strong liking for an item. Another approach is to model ratings as ordered categorical values, enabling the extraction of all possible user ratings. The commonly used set of categories includes "Strongly Disagree", "Disagree", "Neutral", "Agree" and "Strongly Agree". Ratings can also be represented as binary data, where users express their preferences as either liking or disliking an item. Additionally, a specific data type known as unary rating data can be used in recommender systems, where users only have the option to define a like for an item without the ability to specify a disliking. All these different types of data play a vital role in building accurate and effective recommender systems, allowing for the generation of personalized recommendations tailored to individual user preferences.

A. Recommender System Approaches

The definition and the building of recommender systems can be based on several approaches. Three main approaches can be identified: *Collaborative filtering*, *Content-based filtering*, and *Knowledge-based filtering*. The first approach, collaborative filtering, operates on the assumption that users with similar interests may have the same item preferences. Collaborative filtering methods can be roughly divided into two categories : *User-based* and *Item-based* collaborative filtering. The User-based collaborative filtering identifies an ensemble of users who are similar to a target user (i.e. n neighbors). Based on the neighbor's ratings, it predicts the possible rating(s) to a list of products not yet purchased (i.e. product(s) to be recommended) or the top N-recommended products. In this technique, similarity functions are calculated between rows of rating matrix to extract similar users [4]. The second collaborative filtering category, Item-based, tries to extract the highly correlated items (similar items) to the ones that target users are interested in. The extracted items are considered as recommendations. In this category, similarity functions are computed between the columns of the rating matrix, rather than rows, to discover similar items [4].

Concerning the second recommender systems approach, *Content-based filtering*, both user ratings and item properties are used to build recommendations. In this approach, the similarity is calculated based on item features (i.e. product content). Recommended items consist of the filtered list of similar products to those a target user shows high interest. Such type of filter does not suppose that other users must make recommendations for the target user. Based only on what the target user likes, the algorithm will simply pick items with similar contents and recommend them to that user. A content-based filtering system recommends items based on the correlation between the content of the items. This approach is different from the collaborative filtering approach which recommends items based on the correlation between users/items with similar preferences [7].

The third recommender systems approach, knowledge-based, aims to provide recommendations based on product properties, issued by the user, instead of using the history of users' preferences. This approach allows users to explicitly express their desires or explicitly indicate their item preferences before the generation of recommendations. A recommender system is considered "knowledge-based" when it prompts the user to give a series of rules or guidelines on what the results should look like or an example of an item. Then, the system searches through its database of items and returns similar results [8].

B. Related Works: Clustering-Based Recommender Systems

Clustering has been used as a significant technique in the design and the implementation of scalable and efficient recommender systems, known as clustering-based recommender systems. Such systems are based on clustering algorithms to group users or items based on shared preferences or behaviors. The built segments of items or users allow for the recommendation system to focus on specific group characteristics

to improve the computational efficiency and to address data sparsity challenges and cold start problem [9]. For instance, the authors in [10], [11] and [12] used clustering to efficiently recommend news articles based on user interests. The author in [13] leveraged clustering in smart grid applications to enhance recommendations. Ahuja et al. [14] built a Movie recommender system using K-Means clustering and K-Nearest Neighbor. The authors in [15], [16] and [17] utilized clustering in tourism recommendation to group users with similar travel interests.

Recent advancements in clustering-based recommendation systems explore various clustering algorithms and approaches to optimize system performance, user satisfaction, and computational efficiency. Existing methods can be grouped into three main approaches: user-based, item-based, and hybrid.

The first, user-based, groups users with similar behaviors in the same cluster and then recommendations are generated based on local preferences within each cluster. For instance, Yu et al. [18] proposed ClusterFedMet [18], a recommendation algorithm that addresses the challenges of recommendations in federated learning environments. ClusterFedMet improves recommendation accuracy and personalization while keeping data privacy. Additionally, Hosen et al. [19] proposed an optimized recommender system, using PSO-optimized K-means clustering, able to create patient-specific diet recommendations based on historical health data. The proposed system ensures rapid and relevant dietary suggestions for thyroid patients.

On the other hand, the second approach, item-based, allows to group similar items having shared attributes or similar user-interactions[20]. Zhang and Wu [21] developed an e-commerce recommendation model that integrates K-means with a genetic algorithm to enhance recommendation accuracy and speed. The proposed model has a significant boost in item relevance and user satisfaction compared to existing ones. Rather than using a simple clustering method, Airen and Agrawal [22] applied the co-clustering technique in movie recommendations. The fine-tuning of user and movie neighborhood parameters allowed to increase recommendation quality. Their model showed notable improvements in accuracy by leveraging a partitioned clustering approach within large movie datasets.

Concerning the third approach, hybrid clustering, it combines both user and item clustering to simultaneously capture user preferences and item characteristics, leading to a comprehensive recommendation strategy with improved suggestions accuracy and relevance. A recent example of hybrid systems proposed by Wang et al. [23] for tourism recommendation combines multiple recommendation methods, including collaborative filtering and neural networks, to suggest different and relevant tourist spots. A multi-objective approach is used to suggest recommendations that allows to enhance users travel experience. Another hybrid approach, proposed by Iftikhar et al. [24] use bi-clustering within a reinforcement learning framework to adapt recommendations to changing user preferences. This model allows to capture local patterns in user-item interactions by combining bi-clustering with Markov Decision Processes which makes it effective for dynamic recommendation scenarios. Furthermore, another system proposed by Forouzandeh et al. [25], referred to as RESCHet, embeds spectral clustering in heterogeneous networks to analyze relationships in complex item-user networks. The use of spectral clustering to group item-user interactions made it ideal for large datasets where traditional clustering methods may fall short.

All these presented approaches were based on common Clustering algorithms including k-means, hierarchical clustering, density based and Gaussian Mixture Models. The choice of the clustering technique and also the choice of the similarity metric are usually based data characteristics and application requirements. For example, when dealing with sparse datasets, k-means++ or DBSCAN may be favored due to their robustness against sparse data [26]. A good choice of these techniques has a direct impact on the accuracy of the system. Although the effective application of clustering techniques in recommendation systems, conventional techniques may oversimplify the complexity of user preferences by assigning each user or item to a single cluster. However, in real life applications, a more flexible approach is mandatory since users or items may have several preferences and then may belong to multiple clusters. For example, when recommending movies, a user may have interests in both “romantic” and “action” genres of films. This reflects the multifaceted nature of user preferences and the necessity to use more sophisticated and flexible clustering models.

As a solution for the multifaceted nature of user preference, Overlapping clustering methods have proven to be particularly useful in capturing these diverse preferences [27], [28], [20]. Recent works illustrate the advantages of overlapping clustering in various domains. For instance, Heckel and Vlachos [27] introduced a matrix factorization-based approach to identify overlapping co-clusters of clients and products. This method emphasizes interpretability by recommending items that are members of multiple co-clusters. Effectiveness is shown especially in large datasets making it ideal for business-to-business applications. Besides, Vlachos et al. [28] expand the overlapping co-clustering concept to address cold-start problems in matrix factorization. Flexibility of recommendations is improved by allowing users and items to participate in multiple clusters. This flexibility ensures that new users or items are matched effectively which avoids the limitations of traditional clustering methods. Rezghi and Baratnezhad [2] also proposed fuzzy co-clustering methods using non-negative matrix factorization. Overlapping clustering is used to improve the quality of recommendations by reflecting users’ diverse preferences. The proposed approach achieved better sparcity reduction and accuracy compared to non-overlapping methods. Iftikhar et al. [24] also employed overlapping co-clustering within a reinforcement learning framework. By allowing users and items to belong to multiple bi-clusters, the proposed model allowed to effectively capture local patterns and dynamic preferences compared to static collaborative filtering methods, especially in dynamic environments where user preferences may evolve over time. Furthermore, the authors in [29] explored fairness improvement in recommender systems using overlapping clustering. They propose grouping users based on interactions and sensitive attributes, such as age or gender, to provide equitable recommendations across diverse user groups. Overlapping clustering have lead to more balanced outcomes and improved fairness metrics. In the same way, the authors [30] leveraged overlapping clustering to optimize recommendation diversity and novelty. By allowing overlaps, users can explore new items while maintaining access to

familiar favorites. This design has significantly enhanced the diversity of recommendations without compromising precision.

III. Motivation and Research Gap

The design of effective clustering-based recommender systems remains a major research challenge given that these systems must simultaneously ensure accuracy, diversity, interpretability, and computational efficiency. While numerous clustering methods have been proposed, their application to recommender systems still suffer from fundamental limitations that restrict their theoretical and practical contributions.

A first challenge relates to the treatment of user and item clusters as mutually exclusive. Existing clustering-based approaches assume that a user or an item belongs to a single homogeneous group. However, in real-world applications, user preferences are often heterogeneous and multidimensional. For instance, a movie enthusiast may simultaneously enjoy action, drama, and comedy genres, while a book reader may have overlapping interests in historical fiction, biographies, and mystery. Ignoring such overlaps leads to rigid partitions that fail to capture the nuanced and overlapping structures of user preferences. An overlapping co-clustering approach, by contrast, can represent these multifaceted relationships more faithfully and can lead to more personalized recommendations.

A second critical issue concerns interpretability. Recommender systems users usually demand greater transparency regarding the rationale behind generated recommendations. Existing clustering-based methods typically optimize predictive accuracy but fall short in providing explainable justifications for their outputs, particularly when overlapping user–item relationships are involved. Co-clustering offers a promising solution by jointly segmenting users and items. Co-clustering makes it possible to explain recommendations in terms of shared membership across multiple co-clusters which directly improves trust, accountability, and user acceptance.

A third methodological gap concerns the tailoring of recommendation list sizes. Current systems frequently adopt fixed-length recommendation lists, overlooking the dynamic and context-dependent nature of user needs. This "one-size-fits-all" strategy reduces the relevance of recommendations. It becomes possible to adapt list sizes dynamically by integrating the degree of overlap in user–item co-clusters. Such flexibility enables the system to remain context-aware and user-centric, thereby enhancing recommendation utility.

Despite recent advances, existing approaches still exhibit significant limitations. Conventional hard-partition methods remain insufficient to capture multi-faceted user preferences, while soft-clustering approaches provide smoother decision boundaries but lack the structured dual segmentation of co-clustering. Moreover, current methods rarely balance the trade-off between interpretability and predictive performance; most prioritize one dimension at the expense of the other. Finally, although prior studies have begun to explore dynamic list adaptation [2], [30], there is no comprehensive framework that unifies overlap-based co-clustering with modularity maximization, interpretability mechanisms, and personalized list sizing.

To address these gaps, this paper proposes an Overlapping Co-Clustering approach with Modularity Maximization. The framework contributes novel theoretical and practical advancements by: 1) capturing complex overlapping structures of user–item interactions, 2) integrating interpretability into the clustering process to support explainable recommendations, and 3) dynamically adapting list sizes to user contexts. This approach directly responds to the discussed challenges and establishes a more accurate, interpretable, and adaptable recommender system.

IV. Proposed Method: Recommendation based on Overlapping Co-Clustering and Modularity Maximization

The proposed method, called Overlapping Co-clustering with Modularity Maximization (OCCMM), is designed to provide overlapping co-clusters based on modularity maximization using implicit feedback data. The overlapping co-clusters will be then explored to generate flexible and easily interpretable recommendations. The proposed OCCMM method begins by representing the data of the recommender system as a bipartite graph, where users and items are represented as distinct types of vertices. The edges of the graph correspond to the positive feedback given by users on the items. To illustrate the designed OCCMM recommendation process, we have schematized the main steps in Fig. 1. Our proposed method consists of three key steps. First, it begins by building a binary matrix from the bipartite graph. Next, it focuses on creating an ensemble of co-clusters, which will be utilized in the final step to generate recommendations. In the following subsections, we provide a detailed breakdown of the tasks involved in each step of the OCCMM process.

A. Step 1: Data Preprocessing

The data preprocessing step has the objective of effectively transforming the input bipartite graphs of the recommender system into a binary adjacency matrix. Let $G = (N, M, E)$ be the input bipartite graph, where N represents the set of user nodes, M the set of item nodes and E represents the set of edges. The objective is to transform the bipartite Graph G into a binary adjacency matrix $D = (N \times M)$. The value $d_{nm} = 1$ indicates the existence of a relationship (edge in the graph G) between the user node n and the item node m. The adjacency matrix D encompasses all relationships between users and items, serving as implicit feedback data that will be automatically segmented into k overlapping co-clusters to detect similarities between items and users simultaneously. Fig. 1 shows an example of transforming a bipartite graph containing $N = 6$ users and $M = 4$ items into an adjacency matrix $D = (6 \times 4)$ where entries $d_{nm} = 1$ represents the existence of an edge between user n and item m. For instance, the edge between $user_1$ ($n = 1$) and $item_1$ ($m = 1$) indicates that $user_1$ bought $item_1$, prefers it, or any other relevant connection between $user_1$ and $item_1$ resulting in $d_{11} = 1$.

B. Step 2: Generation of Co-Clusters

Given the implicit feedback data matrix $D = (N \times M)$ obtained from the previous step, the objective of the second step is to look for optimal k co-clusters of users $U(N \times k)$ and items $I(M \times k)$ simultaneously based on an iterative modularity-maximization-based process. As described in the

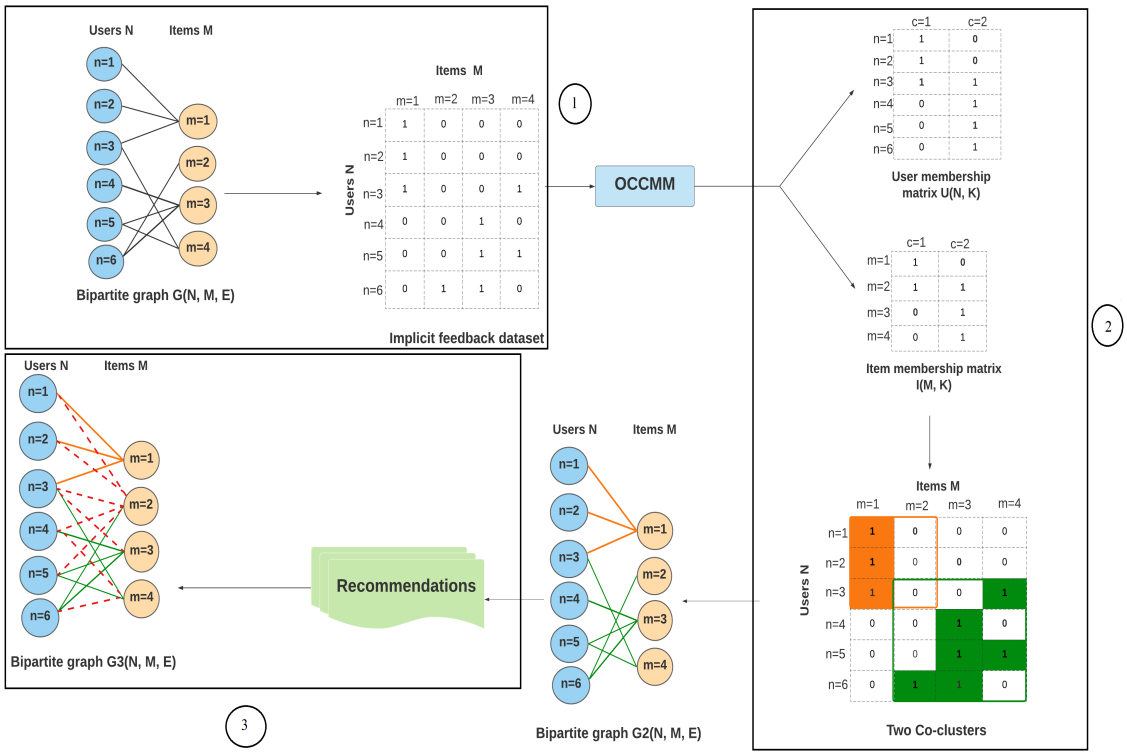


Fig. 1. The designed recommendations steps via the proposed OCCMM method. Step 1 : Data preprocessing, Step 2: generation of co-clusters, Step 3: Generating recommendations using Co-clusters.

works of [27] that uses a modularity maximization of a network for building optimal co-clusters, we define the following modularity objective function for the feedback data:

$$Q(U,I) = \frac{1}{|D|}\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{c=1}^{k}(d_{nm} - \frac{d_n d_m}{|D|})u_{nc}i_{mc} \quad (1)$$
$$-\sum_{n=1}^{N}\sum_{c=1}^{k}u_{nc}\log u_{nc} - \sum_{m=1}^{M}\sum_{c=1}^{k}i_{mc}\log i_{mc}$$

where, $|D| = \sum_n \sum_m d_{nm}$ represents the total number of edges in the network, $d_{nm} \in \{0,1\}$ is an element of the feedback matrix equal to one when a positive feedback of the user n on item m exists, $d_n = \sum_{m=1}^{M} d_{nm}$ and $d_m = \sum_{n=1}^{N} d_{nm}$ represent the degree of user node n and the degree of item node m respectively. The values $u_{nc} \in [0,1]$ and $i_{mc} \in [0,1]$ indicate the fuzzy membership degrees that user n and item m belong to cluster c, respectively. The values $\sum_{n=1}^{N}\sum_{c=1}^{k} u_{nc}\log u_{nc}$ and $\sum_{m=1}^{M}\sum_{c=1}^{k} i_{mc}\log i_{mc}$ denote the separate entropy regularizing users and items membership degree functions separately. The minimization of these two values corresponds to maximizing the fuzzy entropy $-\sum_{n=1}^{N}\sum_{c=1}^{k} u_{nc}\log u_{nc}$ and $-\sum_{m=1}^{M}\sum_{c=1}^{k} i_{mc}\log i_{mc}$.

The maximization problem of the objective function Q respecting to optimal matrices (U) and (I) can be described follows:

max

$$Q(U,I) \quad (2)$$

subject to

$$\sum_{c=1}^{N} u_{nc} = 1, \quad u_{nc} \in [0,1], \quad \forall n = 1,\ldots,N \quad (3)$$
$$\sum_{c=1}^{M} i_{mc} = 1, \quad i_{mc} \in [0,1], \quad \forall m = 1,\ldots,M \quad (4)$$

This problem can be solved through the application of the Lagrange method, involving the utilization of Lagrange multipliers ζ_a and η_b corresponding to constraints (3) and (4) respectively. Eq. (1) can be expressed as:

$$\begin{aligned} Q(U,I) =& \frac{1}{|D|}\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{c=1}^{k}\left(d_{nm} - \frac{d_n d_m}{|D|}\right)u_{nc}i_{mc} \\ &- \sum_{n=1}^{N}\sum_{c=1}^{k}u_{nc}\log u_{nc} - \sum_{m=1}^{M}\sum_{c=1}^{k}i_{mc}\log i_{mc} \\ &+ \sum_{n=1}^{N}\zeta_n\left(\sum_{c=1}^{k}u_{nc} - 1\right) + \sum_{m=1}^{M}\eta_m\left(\sum_{c=1}^{k}i_{mc} - 1\right) \end{aligned} \quad (5)$$

By deriving (Q) with respect to (U) and (I) and setting the gradient to zero, the following expressions are obtained:

$$\frac{\partial Q}{\partial U} = \sum_{m=1}^{M} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) i_{mc} - (1 + \log u_{nc}) + \zeta_n = 0$$
$$\Longleftrightarrow$$
$$u_{nc} = \frac{\exp\left(\sum_{m=1}^{M} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) i_{mc} \right)}{\sum_{c=1}^{k} \exp\left(\sum_{m=1}^{M} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) i_{mc} \right)} \tag{6}$$

$$\frac{\partial Q}{\partial I} = \sum_{n=1}^{N} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) u_{nc} - (1 + \log i_{mc}) + \eta_m = 0$$
$$\Longleftrightarrow$$
$$i_{mc} = \frac{\exp\left(\sum_{n=1}^{N} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) u_{nc} \right)}{\sum_{c=1}^{k} \exp\left(\sum_{n=1}^{N} \left(d_{nm} - \frac{d_n d_m}{|D|} \right) u_{nc} \right)} \tag{7}$$

Eq. (6) and (7) refer to the fuzzy membership of user n to co-cluster c and the fuzzy membership of item m to co-cluster c, respectively. When the membership degrees of user n and item m surpass a threshold θ defined in Eq. (8), u_{nc} and i_{mc} are set to 1, signaling the membership of user n and item m to co-cluster c.

$$\theta = \left(\frac{1}{k} + \frac{1}{\delta \times k} \right) \tag{8}$$

where, k denotes the number of co-clusters and $\delta \in]1, +\infty[$ represents a user-defined parameter introducing flexibility in recommendations. A higher value of δ yields a more extensive list of recommendations, while a lower value restricts the recommendations.

To look for optimal overlapping co-clusters that maximize the overall modularity function, the optimization process starts with random initializations of matrices U and I and an initial modularity score Q_0. Subsequently, a series of operations are iterated by computing new optimal values for user cluster memberships u_{nc} and item cluster memberships i_{mc} using Eq. (6) and (7), respectively. Following this, a new modularity score $Q_t(U, I)$ at iteration t is recalculated based on these updated memberships. This iterative procedure continues until the modularity score remains unchanged or a maximum iteration limit is reached, offering a systematic approach to organizing users and items into fuzzy co-clusters. The final overlapping co-clusters are then built by assigning users and items to similar co-clusters if their membership probabilities surpass a predefined threshold θ. A pseudo-code of the Overlapping Co-clustering via Modularity Maximization algorithm is described in Algorithm 1.

In the context of clustering, modularity allows to measure how well a network is divided into clusters by comparing the actual connections to a randomly connected network. In our context, we aim to find co-clusters of users and items that maximize a certain modularity that ensures users in the same cluster interact with similar items more frequently than a randomly expected one. Based on the adjacency matrix $D(N \times M)$ of users U and items I, the modularity function $Q(U, I)$ evaluates how well users and items are grouped into overlapping co-clusters while maintaining statistical significance. The term $d_{nm} - \frac{d_n d_m}{|D|}$ captures the difference between the actual interactions and the expected random interactions, ensuring that clusters a better meaningful relationships. Entropy terms (log functions) encourage fuzzy membership, indicating that users and items can belong to multiple co-clusters rather than a strict one-to-one assignment.

Algorithm 1 Overlapping Co-clustering via Modularity Maximization

Input: D= Original data matrix, k= number of co-cluster, δ: recommendation flexibility parameter, IterationLimit: Number of maximal iterations

Output: Co-clusters matrices U and I

```
Random initialization of U and I
repeat
  Compute u_nc according to Formula (6)
  Compute i_mc according to Formula (7)
  Compute Q(U, I) using Formula (1) based on u_nc and i_mc
until Unchanged Q(U,I) or IterationLimit
for n = 1 to N do
  for c = 1 to k do
    if u_nc > θ then
      Assign the user n to the co-cluster c
    end if
  end for
end for
for m = 1 to M do
  for c = 1 to k do
    if i_mc > θ then
      Assign the item m to the co-cluster c
    end if
  end for
end for
End
```

C. Step 3: Generating Recommendations using Co-Clusters

The third step is devoted to building recommendations based on the obtained overlapping co-clusters. This step is based on the hypothesis that users and items within the same co-cluster exhibit a strong correlation. By leveraging the relationships identified in the co-clusters, the algorithm generates personalized recommendations for users based on items they have not yet purchased but are correlated within the same co-cluster. An item becomes a potential recommendation for a user when both the user and the item are grouped in the same co-cluster but lack a direct link (e.g. a purchase).

The pseudo-code of generating recommendations using overlapping co-clustering is described in Algorithm 2. Given the input data matrix D and the overlapping co-clusters U and I, the recommendation algorithm examines each user-item pair in the dataset. In cases where the user has not interacted with the item ($d_{nm} = 0$), the algorithm checks if that user and that item belongs to the same co-cluster. If this condition is satisfied, the item is included in the recommendation list for that user.

In Fig. 1, colored squares within co-clusters represent positive or implicit examples, while white squares inside the co-clusters signify recommendations. Analogous to graph theory, the task consists in predicting probable edges in the graph $G(N, M, E)$ for near nodes. The predicted edges, shown as red dotted lines in the $G(N, M, E)$ graph, correspond to user recommendations.

Algorithm 2 Recommendation using Overlapping Co-Clusters

Input: D: Original data matrix, U, I: Overlapping Co-clusters
Output: Users recommendation lists

```
for n = 1 to N do
  for m = 1 to M do
    if d_nm=0 then
      {The user n has not yet purchased the item m}
      for c = 1 to k do
        if u_nc = 1 and i_mc = 1 then
          {Item m and user n are in the same co-cluster}
                        n-RecommendedList ← m
        end if
      end for
    end if
  end for
end for
End
```

D. Computational Effectiveness

The OCCMM computational complexity can be evaluated through the computational complexity of its three main steps: data preprocessing, generation of overlapping co-clusters, and recommendation generation. The computational complexity of the first step, data preprocessing, involves constructing a binary adjacency matrix from the bipartite graph $G(N, M, E)$ and requires iterating over the edges. The computational complexity of this step is approximated by $O(E)$. The second step, co-cluster generation, is the most computationally consuming as it involves iteratively optimizing the user-item matrix based on the proposed modularity-maximization process. The computational complexity of the modularity function requires $O(NM)$ operations per iteration, and updating the membership matrices involves summing over all users and items, leading to a complexity of $O(kNM)$ per iteration, where k is the number of co-clusters. Given that this process is repeated until convergence with a maximal iteration number of t, the total computational complexity of the second step can be evaluated by $O(TkNM)$. Concerning the third step, recommendation generation, involves checking all user-item pairs to determine potential recommendations based on co-cluster memberships, which is approximated by $O(kNM)$. Summing all the computational costs across the three steps, the overall OCCMM computational complexity is evaluated by $O(TkNM)$. In practical applications, this computational complexity can be considered efficient. In real-world applications, the number of iterations T required for convergence is generally small, and the number of co-clusters k is significantly smaller than N and M, which largely reduces excessive computational demands. Consequently, OCCMM can be approximated by $O(NM)$ making it effective even with large-scale recommender systems.

V. Experiments and Empirical Results

A. Datasets Description

To assess the effectiveness of our method, we employed three real-world datasets, each offering distinct types of implicit feedback. The first dataset, from the CiteULike website[1], contains user-curated collections of scientific articles. This dataset comprises $5,551$ users and $16,980$ articles, and the goal is to generate new article recommendations based on these positive examples. In this dataset, recommendations are only based on binary implicit feedback, where each user's collection of articles is treated as positive examples. The data is represented as a binary matrix $D = N \times M$, where $d_{nm} = 1$ indicates that user n has added article m to their collection, and $d_{nm} = 0$ indicates the absence of that article.

The second dataset is a subset of the Last.fm dataset[2], which captures listeners' preferences for artists in a binary form. This dataset consists of $1,226$ listeners and 285 artists and the recommendation task consists of recommending likely preferred artists for each listener based on their implicit feedback. In this dataset, the input matrix $D = N \times M$ has $d_{nm} = 1$ if listener n has shown interest in artist m, and $d_{nm} = 0$ otherwise.

The third dataset is the Movielens dataset which contains $100,000$ movie ratings from 671 users on $9,066$ movies. Users rate movies on a scale of 1 to 5 stars, and we follow the convention from previous research [31], [32] by considering ratings of 3 or higher as positive feedback and ignoring lower ratings. This dataset is also transformed into a binary matrix, with ratings of 3 or above taking a value of 1. The task is to generate movie recommendations based on these positive ratings. Across all datasets, the recommendation problem is modeled as generating new suggestions using implicit feedback, represented consistently in binary matrix form.

Table I summarizes the description of used datasets by giving the number of users, the number of items, and the recommendation task for each dataset.

TABLE I. Datasets Description

Datasets	Users	Items	Recommendation Objective
CiteUlike	5551	16980	Article recommendation
LastFm	1226	285	Music recommendation
Movielens	671	9066	Movie recommendation

The selected datasets were chosen to validate different aspects of our model, including its robustness, ability to model complex user preferences, and performance across varying levels of data density. The CiteULike dataset was chosen for its high sparsity making it a challenging scenario to test OCCMM's ability to handle sparse academic interactions. CiteULike contains approximately 5500 users, 17000 articles,

[1]Launched in October 2004 at the University of Manchester and closed permanently on March 30, 2019.

[2]https://www.last.fm/

and 200000 interactions. The LastFm dataset, containing 1900 users, 17600 artists, and 92800 interactions, was selected due to its representation of diverse musical preferences and multi-genre listening behaviors, allowing for assessing OCCMM's capability to capture overlapping user interests. Finally, we included the Movielens dataset, a relatively denser dataset that enables us to evaluate the scalability and generalizability of OCCMM in less sparse settings.

B. Evaluation Methodology

We assessed the efficacy of the recommendations by measuring Recall and Precision [33]. Fig. 2 schematizes the different steps to calculate Precision and Recall measures to evaluate generated recommendations. The dataset is split into training and testing sets by removing a portion of interactions for each user, simulating unseen data. For instance, with a test set ratio of 0.2, if a user had 10 interactions, 8 were allocated to training, and 2 were reserved for testing. Subsequently, the training data was fed into the OCCMM algorithm to form co-clusters. These co-clusters were then used to generate recommendations. Finally, the test set was utilized to evaluate the recommendation outcomes based on Precision and Recall metrics.

The Precision metric quantifies the ratio of relevant items among the retrieved items, aiming to gauge the proportion of recommended items that are pertinent. It is calculated in Eq. (9) as:

$$Precision\ (n) = \frac{|\{m : d_{nm} = 1\} \cap \{m_1, ..., m_Z\}|}{\{m_1, ..., m_Z\}} \quad (9)$$

where: $|\{m : d_{nm} = 1\}|$ denotes the list of relevant items, $|\{m_1, ..., m_Z\}|$ represents the list of recommended items, and $|\{m : d_{nm} = 1\} \cap \{m_1, ..., m_Z\}|$ represents the list of recommended items that are relevant.

Then, the Recall measure assesses the fraction of relevant items that have been retrieved out of the total relevant items. It measures the proportion of relevant items recommended by the system and is calculated in Eq. (10) as:

$$Recall\ (n) = \frac{|\{m : d_{nm} = 1\} \cap \{m_1, ..., m_Z\}|}{|\{m : d_{nm} = 1\}|} \quad (10)$$

To provide a single balanced score, we build the F_Measure, described in Eq. (11), by computing the harmonic mean of Precision and Recall as follows:

$$F_Measure(n) = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (11)$$

As for precision and Recall, a higher F_Measure value indicates a superior recommendation system. The maximum attainable value for the F_Measure is 1, reflecting perfect precision and recall alignment, while the minimum score is 0, indicating a lack of precision and recall balance.

Furthermore, we assess the Average Recommendation metric (Eq. (12)) that calculates the percentage of recommendations provided for each user, crucial for evaluating recommendations generated from overlapping co-clusters. A high recommendation rate is considered when this value exceeds 50%, indicating that each user receives over half of the recommendations and is computed as:

$$Average_Recommendation = \frac{\sum_{n=1}^{N}\{m_1, ..., m_z\}}{N} \times \frac{1}{M} \quad (12)$$

where, N and M are respectively the total numbers of users and items and $\sum_{n=1}^{N}\{m_1, ..., m_z\}$ represents the total number of recommended items.

C. Number of Co-Clusters and Number of Iterations Parameters

The quality of co-clusters and the convergence of the OCCMM method relies significantly on the pre-determined parameters : size of co-clusters K and the number of iterations. A successful determination of these two parameters can lead to enhanced recommendation quality [27]. Concerning the choice of the optimal number of co-clusters K, we determined this value across various evaluations of co-clusters and modularity values. Fig. 3 gives the obtained scores of modularity by considering different values of co-clusters evaluated on all the three datasets. The plot illustrates distinct modularity patterns across the three datasets (CiteUlike, LastFm, and Movielens) as the number of co-clusters varies. For CiteUlike, the modularity peaks at 0.570 with 22 co-clusters, but it declines beyond this point which suggests overfitting as additional co-clusters are introduced. LastFm shows a more gradual trend, with a peak modularity of 0.312 obtained with 16 co-clusters. Movielens peaks at 0.398 with only 5 co-clusters suggesting that this dataset has fewer distinct clusters. Its modularity decreases rapidly as more co-clusters are added which indicates that the dataset may have a simpler structure and is more sensitive to over-segmentation. All these obtained scores show that all three datasets exhibit a clear peak modularity followed by a decline, reflecting a tendency toward overfitting when the number of co-clusters becomes too large. However, the variation in peak modularity and the number of co-clusters across the datasets highlights differences in their inherent complexity. CiteUlike requires a higher number of co-clusters to capture its structure, whereas Movielens has a much simpler grouping, as indicated by its peak with only 5 co-clusters. This emphasizes the importance of dataset-specific tuning when applying co-clustering methods to maximize modularity.

As for co-clusters, we also studied the influence of the number of iterations parameters in the convergence of the proposed OCCMM method. This study involved evaluating OCCMM modularity across varying number of iterations for CiteUlike, Lastfm, and Movielens datasets. Fig. 4 illustrates the convergence behavior of OCCMM for each dataset over multiple number of iterations. For CiteUlike, the modularity starts close to zero and rapidly increases in the first 10 iterations, with minimal further improvements beyond 10 iterations. For LastFm, the modularity starts similarly near zero but increases more gradually than CiteUlike, eventually

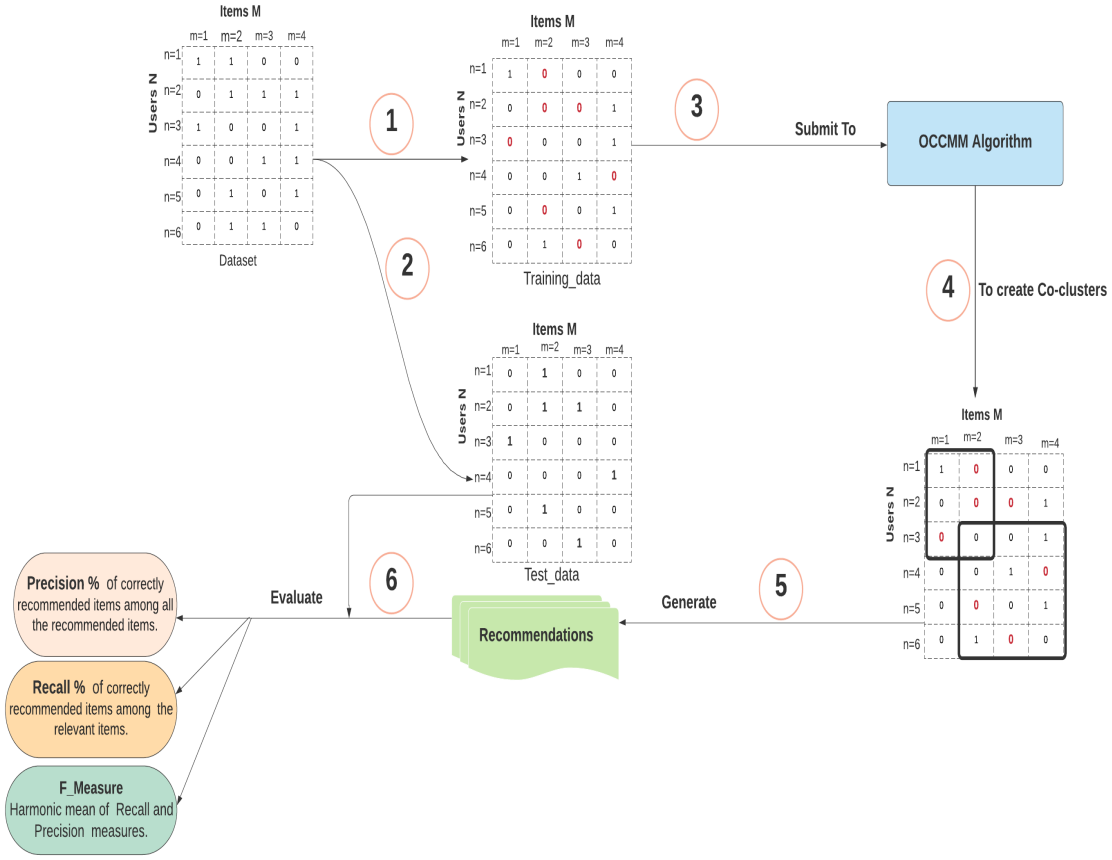


Fig. 2. Evaluation methodology and calculation of evaluation measure based on training and testing datasets.

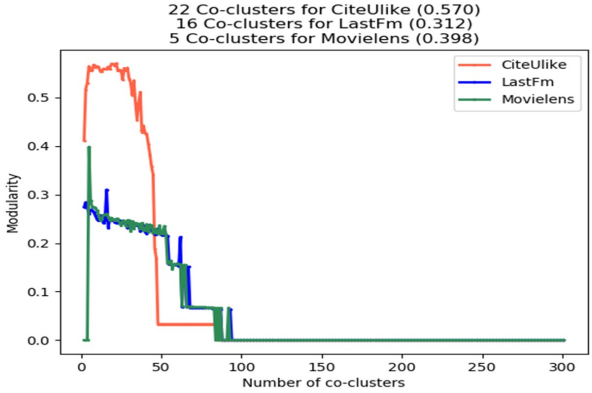


Fig. 3. Maximum modularity values for the CiteUlike, LastFm, and Movielens datasets obtained with OCCMM across varying numbers of co-clusters. The figure shows the optimal number of co-clusters identified for each dataset, along with the corresponding maximum modularity values.

stabilizing around a modularity of 0.3 after 15 iterations. This behavior indicates that OCCMM method finds a good clustering solution earlier with fewer defined or less well-separated clusters compared to CiteUlike, and that additional iterations do not improve the modularity. For Movielens, the modularity rises to about 0.4 after around 18 iterations to be stabilized. This indicates that the dataset has a clear clustering structure that OCCMM quickly identifies good partitionnings, but the co-clustering quality is not as high as in CiteUlike. Across all datasets, we observe that most of the improvement in modularity occurs within the first 10 to 15 iterations, after which the algorithm stabilizes with minimal further gains. For this reason, we set the iterationLimit parameter to 20 for all the further described experiments.

VI. Obtained Empirical Results

We evaluated the effectiveness of the proposed *OCCMM* method in building effective recommendations compared to existing methods in the literature based on *precision*, *recall*, *FMeasure*, and *Average_Recommendation* metrics. Tables II, III, and IV report the obtained values for these measures across different datasets and various methods, including *Bayesian Personalized Ranking (BPR)*, *Fuzzy Co-clustering Modularity Maximization (MMFCC)*, and *Co-clustering via Modularity Maximization (COCLUS)*. These methods are compared with a varying number of co-clusters: $K = 5$, $K = 10$, $K = 16$, and $K = 22$. For our proposed *OCCMM* method, we report results for different values of δ, specifically $\delta = 1$, $\delta = 10^2$, and $\delta = 10^3$.

Concerning CiteUlike dataset, obtained results in Table II show that the proposed **OCCMM** method consistently

TABLE II. OBTAINED PRECISION, RECALL, AND FMEASURE OF OCCMM OF BPR VS MMFCC VS COCLUS VS OCCMM, CITEULIKE DATASET

Datasets	Algorithms	δ value	Precision	Recall	F_Measure	Avg_Recommendation
$K = 5$	BPR		0.014	0.054	0.022	0.058
	MMFCC		0.180	0.755	0.290	0.214
	COCLUS		0.116	0.709	0.199	0.199
	OCCMM	**1**	0.193	0.754	**0.307**	0.212
		10^2	0.171	0.802	0.281	0.243
		10^6	0.171	0.802	0.281	0.244
$K = 10$	BPR		0.014	0.056	0.022	0.058
	MMFCC		0.141	0.683	0.233	0.123
	COCLUS		0.135	0.632	0.222	0.129
	OCCMM	**1**	0.141	0.720	**0.235**	0.136
		10^2	0.134	0.777	0.228	0.172
		10^6	0.132	0.778	0.225	0.178
$K = 16$	BPR		0.014	0.057	0.022	0.058
	MMFCC		0.112	0.664	0.191	0.110
	COCLUS		0.129	0.630	0.214	0.106
	OCCMM	**1**	0.149	0.722	**0.247**	0.130
		10^2	0.136	0.785	0.231	0.170
		10^6	0.135	0.786	0.230	0.171
$K = 22$	BPR		0.014	0.058	0.022	0.058
	MMFCC		0.110	0.713	0.190	0.128
	COCLUS		0.117	0.559	0.193	0.067
	OCCMM	**1**	0.201	0.761	**0.318**	0.138
		10^2	0.195	0.826	0.315	0.188
		10^6	0.162	0.827	0.270	0.189

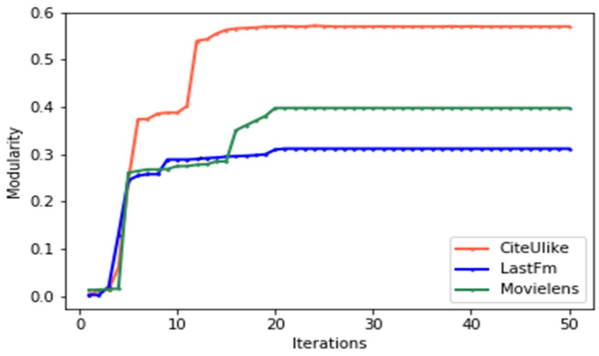


Fig. 4. Maximum modularity values for the CiteUlike, LastFm, and Movielens datasets obtained with OCCMM across varying numbers of iterations. The figure illustrates how the modularity stabilizes after a certain number of iterations for each dataset.

achieves the highest FMeasure values across all K values when $\delta = 1$, highlighting its effectiveness in generating accurate recommendations. For instance, with $K = 22$ and $\delta = 1$, OCCMM achieves an FMeasure of 0.318, largely outperforming both MMFCC and COCLUS having an FMeasure values of 0.190 and 0.193 respectively. This result demonstrates OCCMM superior capability in balancing precision and recall. In terms of recall, OCCMM also performs strongly, especially with larger δ values. With $K = 22$ and $\delta = 10^6$, OCCMM achieves a recall of 0.827, meaning that 82.7% of relevant items are successfully recommended—significantly higher than the recall rates of the other methods. The BPR method achives the loest result eit a recall around 0.058 across all K values, indicating that only 5.8% of relevant items are recommended. However, overlapping methods including MMFCC and COCLUS reach better recall values close to 0.7 meaning that these methods can recommend approximately 70% of relevant items.

Precision values generally range from 0.10 to 0.20 for all methods except BPR, suggesting that, on average, about 10–20 $\%$ of recommendations are relevant to the user. Although OCCMM's precision is comparable to MMFCC and COCLUS, its superior recall and FMeasure values indicate a better overall balance, especially with smaller δ values. The δ parameter in OCCMM also allows for customization of the recommendation list size. For example, with $K = 22$ and $\delta = 1$, OCCMM yields an AverageRecommendation value of 0.138. However, if users prefer a larger recommendation list, setting δ to 10^6 increases the AverageRecommendation to around 0.2, providing more options at the cost of a slight reduction in precision.

For the second dataset, LastFm, the reported results in Table III highlight the competitive performance of OCCMM compared to BPR, MMFCC, and COCLUS in terms of FMeasure and recall metrics across different co-cluster sizes ($K = 5$, $K = 10$, $K = 16$, and $K = 22$). OCCMM achieves the highest FMeasure scores (0.365 and 0.401) for $K = 10$ and $K = 16$ with $\delta = 10^2$. This improvement in FMeasure is driven by a significant increase in recall. OCCMM reaches its highest recall values of 0.628 and 0.729 for $K = 16$ and $K = 22$ with $\delta = 10^2$, indicating that OCCMM can recommend around 70% of relevant items. Additionally, the results demonstrate OCCMM's flexibility in adjusting the recommendation list size based on the configured δ parameter. As δ increases, OCCMM provides larger Average_Recommendation values, reflecting its ability to increase the overlap degree and consequently enlarge

TABLE III. EVALUATION METRICS RESULTS OF BPR VS MMFCC VS COCLUS VS OCCMM, LASTFM DATASET

Datasets	Algorithms	δ value	Precision	Recall	F_Measure	Avg_Recommendation
$K = 5$	BPR		0.012	0.242	0.160	0.175
	MMFCC		0.275	0.481	0.349	0.184
	COCLUS		0.227	0.485	0.309	0.194
	OCCMM	1	0.265	0.457	0.335	0.160
		10^2	0.267	0.556	**0.360**	0.236
		10^6	0.267	0.556	**0.360**	0.237
$K = 10$	BPR		0.013	0.254	0.024	0.175
	MMFCC		0.132	0.413	0.200	0.083
	COCLUS		0.134	0.315	0.188	0.096
	OCCMM	1	0.289	0.440	0.348	0.152
		10^2	0.277	0.535	**0.365**	0.222
		10^6	0.277	0.536	**0.365**	0.223
$K = 16$	BPR		0.013	0.286	0.024	0.175
	MMFCC		0.210	0.552	0.304	0.015
	COCLUS		0.134	0.362	0.195	0.118
	OCCMM	1	0.247	0.398	0.304	0.185
		10^2	0.295	0.626	**0.401**	0.333
		10^6	0.282	0.638	0.391	0.345
$K = 22$	BPR		0.014	0.315	0.026	0.175
	MMFCC		0.163	0.727	**0.266**	0.459
	COCLUS		0.141	0.244	0.178	0.074
	OCCMM	**1**	0.163	0.727	**0.266**	0.459
		10^2	0.163	0.727	**0.266**	0.459
		10^6	0.116	0.759	0.201	0.493

the recommended list of items.

Concerning the Movielens dataset, the results in Table IV illustrate the performance of BPR, MMFCC, COCLUS, and OCCMM across different co-cluster sizes ($K = 5$, $K = 10$, $K = 16$, and $K = 22$). OCCMM consistently demonstrates strong performance in terms of FMeasure, especially for higher values of δ. For instance, OCCMM achieves the highest FMeasure scores of 0.288 and 0.263 for $K = 22$ with $\delta = 1$ and $\delta = 10^2$, respectively. This is attributed to the method's ability to increase recall while maintaining a competitive precision. Notably, OCCMM achieves a recall of 0.639 for $K = 22$ with $\delta = 10^2$ indicating that it can retrieve approximately 64% of relevant items in the recommendation list. The results also reveal OCCMM flexibility in adapting the recommendation list size based on the δ parameter. As δ increases, the Average_Recommendation metric also rises which reflects its ability to adjust the overlap degree and thus increase the number of items recommended to users. This adaptability effectively allows OCCMM to balance precision and recall and offers customizable recommendation lists based on user preference.

Fig. 5 provides insights into the average of users and items within each co-cluster across varying numbers of co-clusters K for three datasets CiteUlike, LastFm, and Movielens obtained using OCCMM with $\delta = 100$. These visualizations reveal distinct patterns in each dataset, which helps understand the optimal number of co-clusters for an effective recommendation. For CiteUlike Dataset which contains 5551 users and 16980 items, Fig. 5(a) shows that obtained results exhibit a large item-to-user ratio per co-cluster at lower values of K. As K increases, this ratio decreases, leading to a more balanced distribution of users and items across co-clusters. According to Table II, partitioning CiteUlike with $K = 22$ achieves a higher FMeasure (0.318) than with $k = 5$, $K = 10$ and $k = 16$. This suggests that $K = 22$ provides an optimal balance between the size of user-item groups and recommendation accuracy. Higher K values help refine co-clusters, focusing more specifically on user-item interactions which can improve recommendation quality by ensuring neither too large nor too small co-clusters. Concerning the LastFm dataset which has 1226 users and 285 items, we observe a different distribution pattern due to the higher density of users compared to items. Based on Table III, the optimal partitioning for LastFm is at $K = 16$, yielding the highest FMeasure (0.401) among the configurations. This suggests that $K = 16$ strikes a good balance, avoiding excessively large or small co-clusters and providing an effective clustering granularity that enhances recommendation accuracy. For the third dataset, Movielens, which contains 671 users and 9066 items, Fig. 5(c) shows a large item-to-user ratio per co-cluster, similar to CiteUlike. Although the highest FMeasure (0.263) is achieved with $K = 22$, as shown in Table IV, a more suitable choice is $K = 5$, where co-clusters provide a balance of neither too dense nor sparse user-item interactions. $K = 5$ ensures sufficient group density while maintaining recommendation quality, avoiding the sparsity that can occur with higher K values.

VII. CONCLUSIONS AND FUTURE WORKS

We proposed in this paper a novel recommender system framework called Overlapping Co-Clustering with Modularity Maximization, designed to address three persistent challenges in clustering-based recommendations: capturing multifaceted user preferences, enhancing interpretability, and adapting recommendation list sizes to individual contexts. The OCCMM

TABLE IV. EVALUATION METRICS RESULTS OF BPR VS MMFCC VS COCLUS VS OCCMM, MOVIELENS DATASET

Datasets	Algorithms	δ value	Precision	Recall	F_Measure	Avg_Recommendation
$K = 5$	BPR		0.041	0.271	0.070	0.110
	MMFCC		0.107	0.506	0.176	0.125
	COCLUS		0.102	0.503	0.169	0.143
	OCCMM	1	0.110	0.492	0.179	0.078
		10^2	0.106	0.566	**0.178**	0.168
		10^6	0.106	0.566	**0.178**	0.169
$K = 10$	BPR		0.139	0.244	0.177	0.175
	MMFCC		0.098	0.485	0.163	0.119
	COCLUS		0.117	0.455	0.186	0.085
	OCCMM	**1**	0.115	0.504	**0.187**	0.084
		10^2	0.112	0.579	**0.187**	0.179
		10^6	0.112	0.581	**0.187**	0.180
$K = 16$	BPR		0.013	0.231	0.024	0.175
	MMFCC		0.192	0.456	**0.304**	0.103
	COCLUS		0.194	0.410	0.263	0.067
	OCCMM	1	0.173	0.506	0.257	0.088
		10^2	0.156	0.598	0.247	0.178
		10^6	0.150	0.598	0.239	0.180
$K = 22$	BPR		0.013	0.231	0.024	0.175
	MMFCC		0.170	0.528	0.257	0.130
	COCLUS		0.195	0.401	0.262	0.065
	OCCMM	**1**	0.194	0.562	**0.288**	0.117
		10^2	0.166	0.639	0.263	0.198
		10^6	0.166	0.639	0.263	0.198

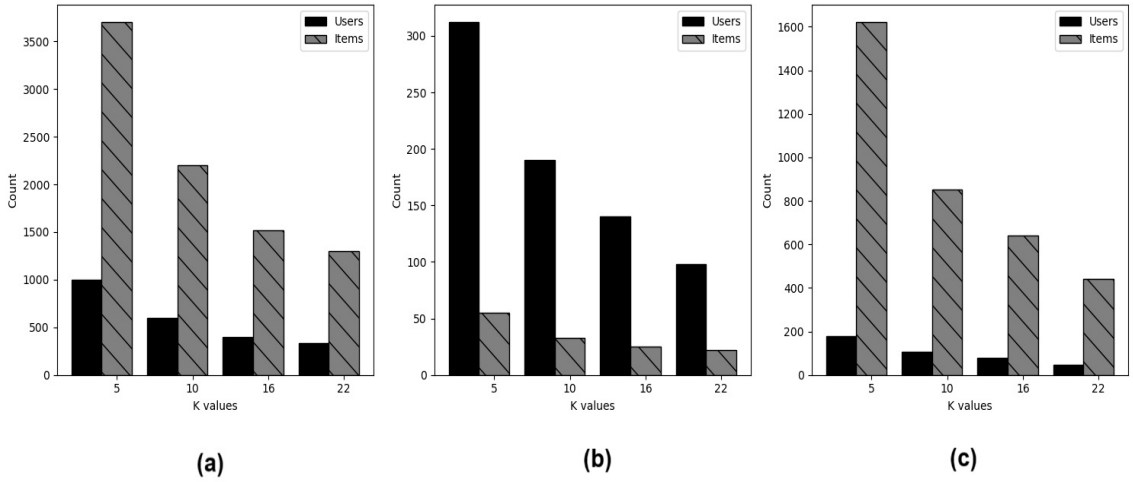


Fig. 5. Average number of Users and Items per Co-cluster across varying K obtained with OCCMM with $\delta = 100$ on: (a) CiteUlike dataset (b) LastFm dataset (c) Movielens dataset.

approach operates in three key steps: 1) implicit data preprocessing, 2) construction of overlapping co-clusters through modularity maximization, and 3) recommendation generation based on these co-clusters. This design effectively balances accuracy, interpretability, and adaptability. By modeling overlapping user–item relationships, OCCMM allows for capturing complex and realistic preference structures. Additionally, it provides transparent rationales for recommendations by using co-clusters rather than simple clusters and also ensures responsiveness to diverse user needs through dynamically adjusting list sizes. Experimental results on real-world datasets confirmed that OCCMM outperforms existing methods in terms of both accuracy and personalization.

While OCCMM demonstrates strong empirical performance, several limitations should be acknowledged. First, the current framework does not explicitly integrate continuous user feedback (implicit or explicit), which may limit its ability to iteratively refine recommendations. Second, the scalability of OCCMM under very large-scale datasets remains an open challenge. Computational complexity could be mitigated through advanced optimization or parallelization strategies. Third, the model is currently evaluated in static environments, whereas many real-world systems require realtime adaptation to dynamic user behaviors. Finally, although OCCMM balances

interpretability and accuracy, its explainability mechanisms are limited to cluster-level reasoning, which may not always capture fine-grained user justifications.

The effectiveness of OCCMM can be improved in future works by investigating other techniques to address critical issues related to recommendation systems. First, the incorporation of explicit and implicit user feedback into the co-clustering process may enable the system to refine recommendations based on user interactions. In addition, optimization techniques can be investigated to improve the scalability of OCCMM when handling large-scale datasets. One can also investigate the integration of OCCMM with other recommendation techniques, such as deep learning or collaborative filtering, to further improve accuracy and robustness. Furthermore, a promising direction to improve the accuracy and robustness of OCCMM is to integrate deep learning-based clustering which can give more accurate results. One can investigate the implementation of real-time recommendation mechanisms within OCCMM to ensure instant adaptation to evolving user behaviors and preferences in live environments.

ACKNOWLEDGMENT

This work was funded by the University of Jeddah, Jeddah, Saudi Arabia, under grant No. ($UJ - 22 - DR - 48$). The authors, therefore, acknowledge with thanks the University of Jeddah for its technical and financial support.

REFERENCES

[1] J. Das, P. Mukherjee, S. Majumder, and P. Gupta, "Clustering-based recommender system using principles of voting theory," in *2014 International Conference on Contemporary Computing and Informatics (IC3I)*. IEEE, 2014, pp. 230–235.

[2] M. Rezghi and E. Baratnezhad, "A novel fuzzy co-clustering method for recommender systems via inverse stereographic nmf," *Expert Systems with Applications*, vol. 259, p. 125301, 2025.

[3] T. George and S. Merugu, "A scalable collaborative filtering framework based on coclustering," in *Fifth IEEE International Conference on Data Mining (ICDM 05)*. IEEE, 2005, pp. 4–11.

[4] C. C. Aggarwal, J. L. Wolf, P. S. Yu, C. Procopiuc, and J. S. Park, "Fast algorithms for projected clustering," *SIGMOD Rec.*, vol. 28, no. 2, p. 61–72, Jun. 1999.

[5] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 2015, pp. 1235–1244.

[6] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *information retrieval*, vol. 4, no. 2, pp. 133–151, 2001.

[7] J. Son and S. B. Kim, "Content-based filtering for recommendation systems using multiattribute networks," *Expert Systems with Applications*, vol. 89, pp. 404–412, 2017.

[8] J. Wu, "Knowledge-based recommender systems: An overview," Oct 2019, last checked on Aug 22, 2020.

[9] Y. Pérez-Almaguer, R. Yera, A. A. Alzahrani, and L. Martínez, "Content-based group recommender systems: A general taxonomy and further improvements," *Expert Systems with Applications*, vol. 184, p. 115444, 2021.

[10] S. Cleger-Tamayo, J. M. Fernández-Luna, J. F. Huete, R. Pérez-Vázquez, and J. C. Rodríguez Cano, "A proposal for news recommendation based on clustering techniques," in *Trends in Applied Intelligent Systems*. Springer, 2010, pp. 478–487.

[11] D. Z. Ulian, J. L. Becker, C. B. Marcolin, and E. Scornavacca, "Exploring the effects of different clustering methods on a news recommender system," *Expert Systems with Applications*, vol. 183, p. 115341, 2021.

[12] N. S. Diana Sousa and L. Spagnolo, "Recommending news articles for public health intelligence," in *The 6th Workshop on Health Recommender System in the 18 th ACM Conference on Recommender Systems (HealthRecSys'24)*, Bari, Italy. ACM, 2024.

[13] N. P. Chadalavada and N. Ramachandran, "An efficient neighbor nodes trust tagging model for optimized route detection in smart grids with secure data transmission," *Heliyon*, vol. 10, no. 11, p. e32074, 2024.

[14] R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbor," in *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE, 2019, pp. 263–268.

[15] F. Li and C. Zhang, "A deep learning-based recommender model for tourism routes by multimodal fusion of semantic analysis and image comprehension," *Journal of Circuits, Systems and Computers*, 2024.

[16] R. Cao, Z. Li, P. Wei, Y. Tian, and C. Zheng, "A hybrid tourism recommendation system based on multi-objective evolutionary algorithm and re-ranking," in *Advanced Intelligent Computing Technology and Applications*. Singapore: Springer Nature Singapore, 2023, pp. 363–372.

[17] S. Wang, R. Cao, Y. Tian, and C. Zheng, "Hybrid tourism recommendation system: A multi-objective perspective," in *2022 IEEE Congress on Evolutionary Computation (CEC)*. IEExplore, 2022, pp. 1–8.

[18] E. Yu, Z. Ye, Z. Zhang, L. Qian, and M. Xie, "A federated recommendation algorithm based on user clustering and meta-learning," *Applied Soft Computing*, vol. 158, p. 111483, 2024.

[19] M. A. Hosen, S. H. Moz, S. S. Kabir, D. S. M. Galib, and D. M. N. Adnan, "Enhancing thyroid patient dietary management with an optimized recommender system based on pso and k-means," *Procedia Computer Science*, vol. 230, pp. 688–697, 2023, 3rd International Conference on Evolutionary Computing and Mobile Sustainable Networks (ICECMSN 2023).

[20] A. M. A. Al-Sabaawi, H. Karacan, and Y. E. Yenice, "A novel overlapping method to alleviate the cold-start problem in recommendation systems," *International Journal of Software Engineering and Knowledge Engineering*, vol. 31, no. 09, pp. 1277–1297, 2021.

[21] W. Zhang and Z. Wu, "E-commerce recommender system based on improved k-means commodity information management model," *Heliyon*, vol. 10, no. 9, p. e29045, 2024.

[22] S. Airen and J. Agrawal, "Movie recommender system using parameter tuning of user and movie neighbourhood via co-clustering," *Procedia Computer Science*, vol. 218, pp. 1176–1183, 2023, international Conference on Machine Learning and Data Engineering.

[23] Z. Wang, J. Chen, J. Li, and Z. Wang, "Interest community-based recommendation via cognitive similarity and adaptive evolutionary clustering," *Chaos, Solitons and Fractals*, vol. 185, p. 115085, 2024.

[24] A. Iftikhar, M. A. Ghazanfar, M. Ayub, S. Ali Alahmari, N. Qazi, and J. Wall, "A reinforcement learning recommender system using bi-clustering and markov decision process," *Expert Systems with Applications*, vol. 237, p. 121541, 2024.

[25] S. Forouzandeh, K. Berahmand, R. Sheikhpour, and Y. Li, "A new method for recommendation based on embedding spectral clustering in heterogeneous networks (reschet)," *Expert Systems with Applications*, vol. 231, p. 120699, 2023.

[26] R. D. and D. M., "A systematic review and research perspective on recommender systems." *J Big Data*, vol. 9, 2022.

[27] R. Heckel, M. Vlachos, T. Parnell, and C. Dünner, "Scalable and interpretable product recommendations via overlapping co-clustering," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*. IEEE, 2017, pp. 1033–1044.

[28] M. Vlachos, C. Mendler-Dünner, R. Heckel, V. Vassiliadis, T. Parnell, and K. Atasu, "Addressing interpretability and cold-start in matrix factorization for recommender systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, pp. 1–1, 05 2018.

[29] Y. Deldjoo, V. W. Anelli, H. Zamani, A. Bellogín, and T. D. Noia, "A flexible framework for evaluating user and item fairness in recommender systems," *User Model. User Adapt. Interact.*, vol. 31, no. 3, pp. 457–511, 2021.

[30] C. E. Berbague, N. E. islem Karabadji, H. Seridi, P. Symeonidis, Y. Manolopoulos, and W. Dhifli, "An overlapping clustering approach for precision, diversity and novelty-aware recommendations," *Expert Systems with Applications*, vol. 177, p. 114917, 2021.

[31] W. Pan and L. Chen, "Gbpr: Group preference based bayesian personalized ranking for one-class collaborative filtering," in *Twenty-Third International Joint Conference on Artificial Intelligence*. IJCAI, 2013.

[32] V. Sindhwani, S. Bucak, J. Hu, and A. Mojsilovic, "A family of non-negative matrix factorizations for one-class collaborative filtering problems," in *Proceedings of the ACM Recommender Systems Conference, RecSys*. ACM, 2009.

[33] G. Schröder, M. Thiele, and W. Lehner, "Setting goals and choosing metrics for recommender system evaluations," in *UCERSTI2 workshop at the 5th ACM conference on recommender systems, Chicago, USA*, vol. 23. ACM, 2011, p. 53.

Applying a Lightweight Graphics Library to Visually Corroborate Learning in Programming Introduction Courses

Claudia De La Fuente[1], Cristian Vidal-Silva[2], Liza Jegó-Mendoza[3], Patricia Pedrero-Valenzuela[4]
Facultad de Ingeniería, Universidad de Talca, Talca, Chile[1,2,3]
Departamento de Administración de Educación Municipal, Talca, Chile[4]

***Abstract*—Learning to program in first-year courses is challenging because the link between source code and program behaviour is not immediately visible to novices. This paper reports on the deployment of UFramework, a lightweight graphics library developed in C++/Visual Studio, designed to help students visually corroborate their learning by observing the on–screen effects of their own algorithms. The experience was implemented in a Structured Programming module that combines lectures, labs, and project-based assignments. We 1) describe the design principles and architecture of the library, 2) present a portfolio of progressively scaffolded assignments (shooter prototype, grid map parsing, spatial quadrants), 3) outline the assessment rubric and its alignment with intended learning outcomes, and 4) report multi-year descriptive evidence that includes pass rates and qualitative reflections. Results show improved student engagement and higher pass rates in the most recent cohorts, together with qualitative evidence of increased motivation and clearer problem decomposition. While the findings are limited to a single institution and remain descriptive, they suggest that lightweight, visual-first workflows can lower barriers to learning programming and foster computational thinking competencies such as decomposition, abstraction, algorithmic design, and debugging. Future work should include controlled comparisons and broader validation to strengthen internal validity and explore the applicability of this approach to non-visual domains.**

Keywords*—Structured programming; visual corroboration; self-regulated learning; metacognition; program visualization; project-based learning

I. Introduction

Learning to program in first-year courses is challenging because the relationship between source code and program behaviour is not immediately visible to novices. Without timely, interpretable feedback, misconceptions persist and motivation declines [1], [2], [3]. To reduce this gap, we deployed UFramework, a lightweight graphics library in C++ (Visual Studio toolchain) that allows students to see the effect of loops, conditionals, arrays, and simple state machines with minimal setup. The design intention is to preserve working memory for algorithmic reasoning while delivering immediate, motivating, on-screen feedback [4], [5].

In our institution, the intervention was embedded in a first-year Structured Programming module that combines lectures, lab sessions, and project-based assignments. Students implemented small visual programs and documented decisions in short reflective notes. Fig. 2 provides an overview of the dominant terms appearing in reflections and forum posts—learning, students, programming, education—which is coherent with the course aims.

1) Rationale: Two strands of previous work underpin the approach. First, program visualization and animation help externalize state changes and support comprehension for novices [1], [2]. Second, designs that minimize extraneous cognitive load and support self-regulated learning enable students to devote resources to core schemas and problem solving [4], [5]. We also draw on project-based learning (PBL) to sustain engagement and cultivate productive habits for introductory programming [6], [7], and we align activities with competencies commonly associated with computational thinking [8], [9].

2) Objectives and research questions: The study seeks practical improvements in introductory programming while documenting the design so others can reuse it. We address the following questions:

RQ1. Does a lightweight, visual workflow help students connect code with behaviour in early assignments? RQ2. How do scaffolded visual tasks map to intended learning outcomes (decomposition, abstraction, testing)? RQ3. What descriptive changes are observable in pass rates and student reflections across cohorts?

TABLE I. From Challenges to Design Decisions. The Mechanisms are Intentionally Minimal to Reduce Setup Cost and Cognitive Overhead

Challenge	Design mechanism	Intended effect
Invisible program state	On-screen visual outcomes for each step	Comprehension
High setup overhead	Small, discoverable API and starter templates	Time on task
Early frustration	Short, game-flavoured tasks with rapid feedback	Motivation
Shallow assessment	Rubric beyond correctness (decomposition, docs)	Quality

As shown in Table I, each design decision was mapped to a specific instructional challenge. This mapping ensured that mechanisms such as minimal APIs or short tasks directly addressed barriers faced by novices.

3) Contributions: This study 1) details the library's design principles and conceptual architecture, 2) presents a portfolio of progressively scaffolded assignments (shooter prototype, grid map parsing, spatial quadrants), 3) outlines the assessment rubric and alignment with the intended learning outcomes, and 4) reports multi-year descriptive evidence that includes pass

rates and qualitative themes [8], [9], [3]. The remainder of the paper is organized as follows: Section II introduces the theoretical framework; Section III presents materials and methods; Section IV reports results; Section V discusses implications and limitations; and the paper closes with conclusions and future work.

II. Theoretical Framework

This work is based on four strands. Program visualization externalizes otherwise invisible state changes and has been shown to aid comprehension in introductory contexts [1], [2], [3]. Second, designs that explicitly limit extraneous cognitive load and structure the learning process can free resources for core schema acquisition and problem solving; we draw on work that integrates cognitive load theory with usability engineering as well as evidence on self-regulated learning in computing education [4], [5]. Third, we leverage project-based learning (PBL) to sustain engagement and create concrete artefacts early; prior studies report gains when PBL is tied to novice programming (often with block-based or visual-first tasks) [6], [10], [11], [7]. Finally, the activities are aligned with competencies frequently associated with computational thinking (CT), including decomposition, abstraction, and algorithmic design [8], [9], [12].

Fig. 1 summarizes the theoretical stance. Visualization provides concrete, inspectable outputs for novice code; PBL supplies authentic tasks and milestones; cognitive load and SRL practices (e.g. short reflective notes, explicit planning and monitoring prompts) target process quality; and CT offers a language to articulate skills beyond surface correctness.

Table II maps each construct to concrete mechanisms used in the course and points to representative references in the CS education literature. Together, these strands motivate the design choices reported in the following section.

A. Related Work

Previous research on lightweight visualization tools has highlighted persistent challenges in helping novices connect code and behavior [1], [13]. Although project-based learning studies demonstrate improvements in engagement and competence [6], [11], fewer reports describe lightweight, domain-specific graphics libraries integrated into CS1 contexts. More recent contributions emphasize the need for controlled comparisons and greater validation [10], [7]. Our work contributes to this landscape by documenting a reusable design, multi-year implementation, and descriptive evidence that complements these existing strands.

III. Materials and Methods

A. Context and Participants

The intervention was carried out on a first-year *Structured Programming* course at a Chilean university. The module combines weekly lectures, lab practice, and project-based assignments. The cohorts span terms 2018–1 to 2021–1. The instructional approach emphasizes visible program behaviour for novices, drawing on evidence from program visualization and novice-facing tools [1], [2], [3] and from project-based learning (PBL) applied to introductory programming [6], [10], [11], [7].

Fig. 3 summarises the structure of the intervention, where UFramework is embedded in the Structured Programming course through visual assignments and assessments. This integration was designed to provide immediate feedback while also producing multi-year evidence on student performance and engagement.

B. The UFramework Library

UFramework is a lightweight C++ library (Visual Studio toolchain) exposing a minimal, discoverable API organized into four groups: Core (initialization, main loop), Drawing (sprites, shapes, text), Input (keyboard), and Time (ticks, fixed-step). The design goal is to minimize setup and ceremony so that learners can move quickly from code to on-screen behaviour, aligning with recommendations to reduce extraneous cognitive load and support self-regulated practices [4], [5].

The overall architecture is summarised in Fig. 4, which highlights the modular organisation of UFramework into core, drawing, input, and time components.

C. Assignments and Scaffolding

The students completed three visual assignments aligned with the CS1 syllabus and with competencies typically associated with computational thinking (CT)—decomposition, abstraction, algorithmic design [8], [9], [12].

- A1: Shooter prototype (arrays, collisions, fixed time step). Fig. 5 illustrates the core entities.
- A2: Grid map parsing (file I/O, nested loops, index arithmetic). Fig. 6 shows a sample input/output.
- A3: Spatial quadrants (coordinate transforms, conditionals). Fig. 7 depicts the target outcome.

D. Learning Outcomes and Alignment

The intended learning outcomes (ILOs) emphasize decomposition, abstraction, and testing (aligned with CT [8], [12]). Table III maps the ILOs to the assignments and assessment criteria.

E. Study Timeline and Data Collection

The study was carried out over multiple terms with common milestones: library onboarding, A1–A3 delivery, and final exam. Pass rates were collected from course records; qualitative reflections were gathered from brief lab notes and forum posts. Fig. 11 summarizes the sequence.

Table VI outlines a structured roadmap for extending this work. It highlights planned studies and artefacts, ranging from controlled comparisons to instrument development and cross-course tracking, together with their primary measures and rationales. This matrix provides concrete directions for strengthening internal validity and exploring the generalisation of lightweight, visual-first workflows.

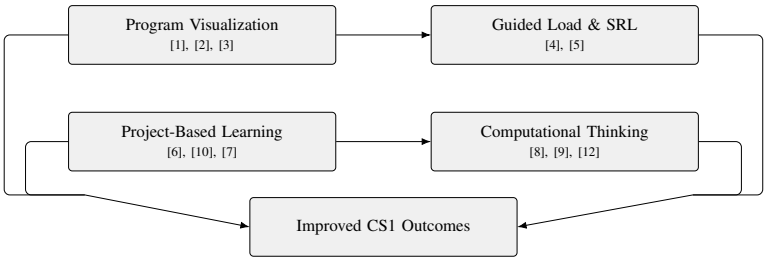
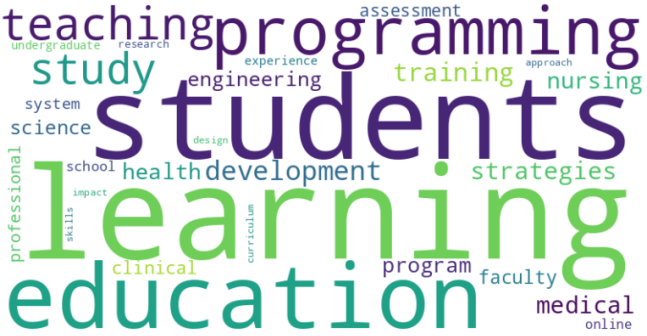


Fig. 1. Conceptual model: program visualization, guided cognitive load with self-regulated learning (SRL), and PBL—framed by CT—contribute to improved introductory programming outcomes.

TABLE II. CONSTRUCTS, ROLES IN THE STUDY, OPERATIONALISATION IN OUR COURSE, AND REPRESENTATIVE EVIDENCE FROM THE LITERATURE

Construct	Role in this study	Operationalisation	Representative evidence
Program visualization	Make program state observable	Visual outcomes for loops, arrays, state machines; stepwise checks	[1], [2], [3]
Cognitive load / SRL	Reduce extraneous load; support self-regulation	Minimal-ceremony API; templates; short planning/monitoring/evaluation notes	[4], [5]
Project-based learning	Sustain engagement; authentic artefacts	Small, visual projects with rapid feedback and clear milestones	[6], [10], [7], [11]
Computational thinking	Frame target competencies	Decomposition, abstraction, algorithmic design emphasised in tasks & rubric	[8], [9], [12]



(a) Lab reflections.



(b) Course forum.

Fig. 2. Frequent terms used by students during the semester. These visuals offer a quick overview and complement, rather than replace, formal qualitative coding.

TABLE III. ALIGNMENT OF INTENDED LEARNING OUTCOMES (ILOS) WITH ASSIGNMENTS AND RUBRIC CRITERIA

ILO	Assignment focus	Rubric criteria (excerpt)
Decomposition	Split responsibilities into functions; manage state	Decomposition; Correctness
Abstraction	Generalize drawing/input patterns; reuse helpers	Use of API; Documentation
Testing & Debugging	Stepwise checks, input validation, edge cases	Correctness; Playability

F. Instruments and Metrics

Assessment combined automated checks (public test sets) and human grading via a rubric. Short reflective notes scaffolded planning, monitoring, and evaluation activities associated with self-regulated learning [5]. Quantitative metrics included pass rates and completion rates for A1–A3; qualitative coding focused on themes related to visibility of program state and motivation. The approach is consistent with visual-first teaching resources in the novice space [9] and systematic accounts of program visualization [1].

G. Ethical Considerations

All analyzes used de-identified cohort summaries (no individual grades reported). Student reflections were coded at aggregate level. The study follows the standard educational practice guide for the improvement of routine courses.

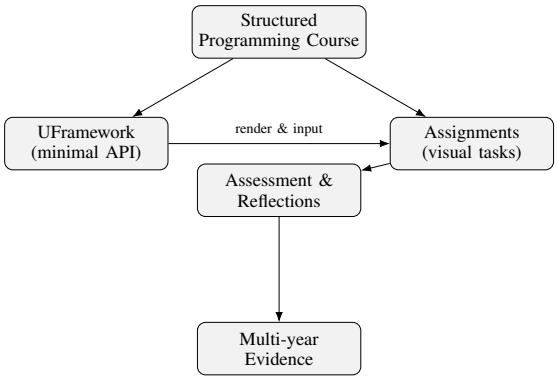


Fig. 3. Overview of the teaching intervention: the course integrates a minimal graphics library with visual assignments and structured assessment to generate multi-year evidence.

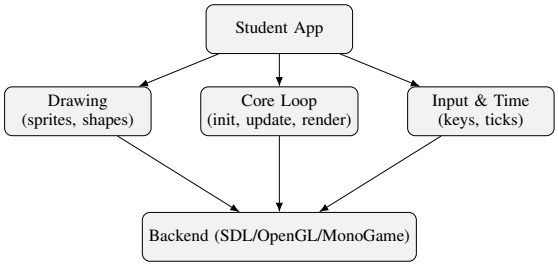


Fig. 4. Library overview. A small set of modules provides immediate visual feedback while abstracting platform details.

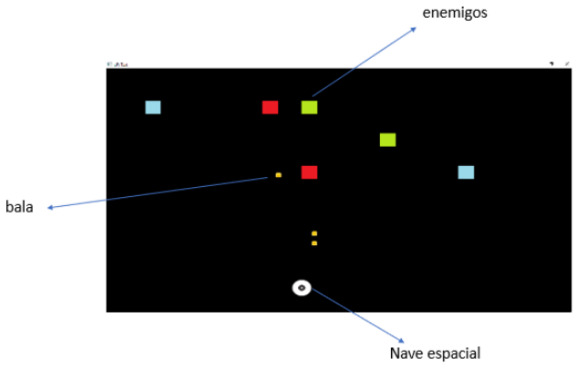


Fig. 5. A1 (Shooter): students implement player movement, enemy spawn, and simple bounding-box collisions using arrays and a fixed time step.

IV. Results

A. Quantitative Outcomes

Fig. 8 summarises pass rates by term for the introductory programming course. After the lightweight library and scaffolded visual assignments became the backbone of the course, the highest pass rate appears in 2021–1. While these results are descriptive, they are consistent with the direction reported by studies that link project-based learning and novice-friendly visual workflows with improved engagement and performance [6], [10], [11], [7] and with accounts of program visualization supporting novice comprehension [1], [2], [3].

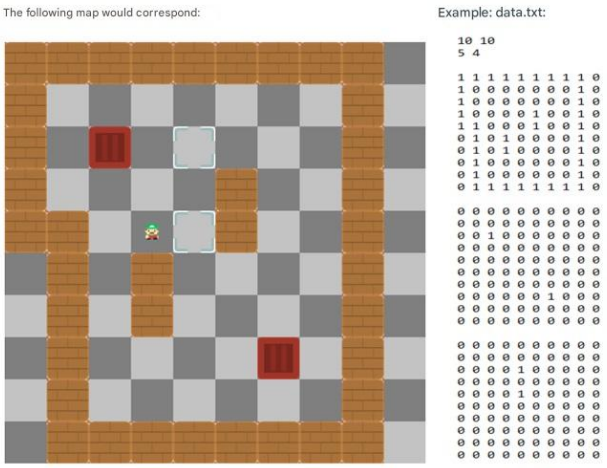


Fig. 6. A2 (Grid map): A textual map is parsed and rendered as a grid-based scene, reinforcing loops, arrays, and file I/O.

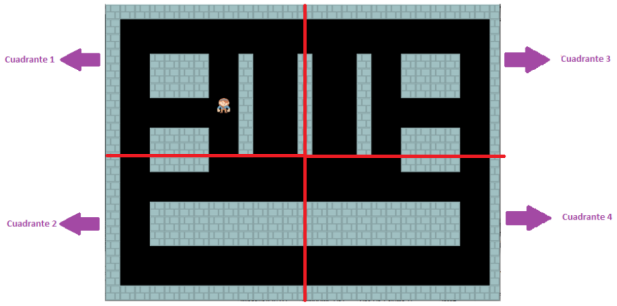


Fig. 7. A3 (Quadrants): Learners compute axis lines and classify entities by quadrant, practicing decomposition and coordinate reasoning.

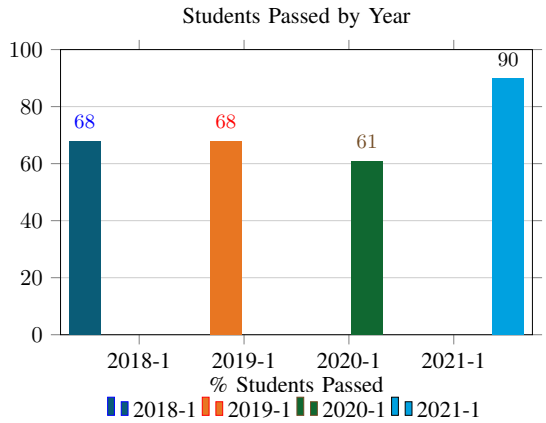


Fig. 8. Pass rates by term (2018–1, 2019–1, 2020–1, 2021–1). Labels indicate the percentage of students who passed each term.

Table IV reports the same percentages as Fig. 8. Enrollment counts are not disclosed in this report; the analysis focuses on cohort-level rates.

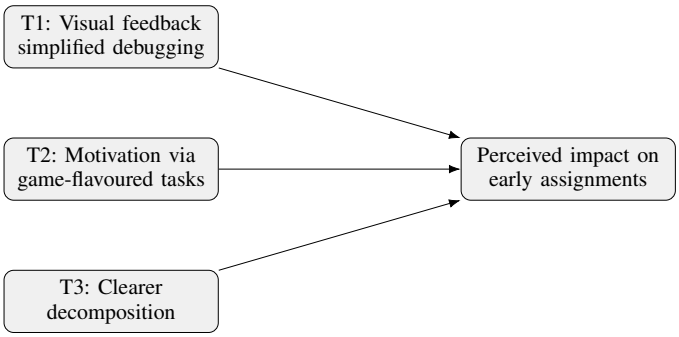


Fig. 9. Thematic summary of qualitative reflections across cohorts. Themes align with prior work on program visualization and structured project tasks for novices [1], [6], [7].

TABLE IV. COHORTS AND PASS RATES (PERCENTAGES AS IN FIG. 8; ENROLLMENT COUNTS NOT DISCLOSED)

Term	Enrolled	Pass rate (%)
2018–1	*n/a*	68
2019–1	*n/a*	68
2020–1	*n/a*	61
2021–1	*n/a*	90

B. Qualitative Findings

Student reflections (short lab notes and forum posts) were coded for recurring themes relevant to early programming. Three dominant themes emerged: (T1) visual feedback simplified debugging; (T2) motivation increased with game-flavoured tasks; and (T3) clearer decomposition of problems. These themes cohere with evidence that visual-first activities help novices externalise program state [1], [2] and that structured, project-based tasks can support persistence and productive study habits [6], [7], [5]. Fig. 9 provides a schematic summary.

To complement the thematic map, the earlier word clouds (Fig. 2) offer a coarse overview of the salient terms (learning, students, programming, education), which is consistent with the course objectives and the emphasis on visual-first [9].

C. Robustness and Reporting

Across terms, assignments shared a common rubric and public test sets, and the library onboarding sequence remained stable. Design choices (minimal ceremony, visible results) target a reduction in extraneous cognitive load and encourage self-regulated practices (brief planning/monitoring notes), which the literature associates with improved process quality for novices [4], [5]. Given the observational nature of the data, we refrain from causal claims; Section V discusses limitations and avenues for controlled comparisons.

1) Validation measures: Although pass rates offer an accessible metric, stronger validation is needed. Future iterations of the course will incorporate concept inventories aiming at tracing and state reasoning [14], as well as controlled comparisons between lightweight and engine-based workflows. These measures will enhance internal validity and allow causal attribution.

To further strengthen the robustness of our findings, we elaborate on three complementary aspects:

- The validation measures that will enhance internal validity,
- The transparency of the qualitative coding process, and
- The breadth of computational thinking competencies addressed by the assignments.

These elements provide additional methodological clarity and situate our descriptive evidence within a broader research context.

2) Qualitative coding: Student reflections were independently coded by two researchers using inductive thematic analysis. The agreement was quantified using Cohen's kappa ($\kappa = 0.81$), indicating substantial reliability. Discrepancies were resolved through discussion until consensus was reached. This process increases the transparency and robustness of the qualitative findings.

3) Computational thinking competencies: Although decomposition and abstraction emerged most prominently, tasks also reinforced algorithmic design and debugging strategies. For example, the Shooter assignment required iterative refinement of collision logic, while the Grid Map task emphasized input validation and error handling, both aligning with CT competencies.

V. DISCUSSION

Our descriptive results are consistent with a long-standing observation in CS education: what most affects learning with visualizations is how learners engage with them rather than what the visualization shows [15], [16]. By making program state and control flow immediately visible, UFramework helps novices connect code with behaviour. This aligns with evidence on program visualization tools and environments that are designed for novices [1], [2], [3], [13]. Complementarily, keeping the API minimal and templates ready-to-run reduces extraneous cognitive load and supports self-regulated practices [4], [5].

1) Interpretation relative to prior work: The course design aligns with two complementary perspectives. First, classic reviews of learning-to-program highlight persistent novice difficulties (e.g. tracing, mental models, and mapping between syntax and execution) and recommend explicit scaffolding and externalizations of run-time behavior [14]. Second, research on lowering barriers for beginners emphasises designs that foreground meaningful outcomes while minimising incidental complexity [17]. Our assignments instantiate both: visual artifacts keep the attention on behavior, while the lightweight library avoids the ceremony of full-fledged engines, similar to the rationale behind widely adopted visualizers such as Python Tutor [13].

2) Implications for course and tool design: Two levers seem particularly actionable (Fig. 10): 1) design levers in the tool (fast path from code to observable behaviour; single, predictable update loop; fixed time-step utilities), and 2) curricular levers (short, visual tasks with rapid feedback; rubric criteria beyond correctness; metacognitive prompts). These choices are consistent with the program visualization literature [16], [1] and with PBL reports indicating gains in engagement and

TABLE V. PRACTICAL TACTICS, RATIONALES, AND WHERE THEY FIT IN THIS COURSE

Tactic	Rationale (evidence)	Placement
Single update loop template	Reduces incidental complexity; supports tracing ([14])	Starter code
Stepwise visual checks	Engagement with visualization matters ([15], [16])	A1–A3 labs
Minimal, discoverable API	Lowers barriers for novices ([17])	Library design
Metacognitive micro-notes	Supports SRL and process quality ([5])	Post-lab
Short visual projects	PBL linked to motivation/competence ([6], [7])	Weekly tasks

TABLE VI. FUTURE WORK MATRIX: STUDIES, DESIGNS, MEASURES, AND RATIONALE

Study / Artefact	Design	Primary measures / Rationale
Controlled comparison: lightweight vs. engine-based	Parallel sections; common syllabus	Concept inventory (tracing/state), assignment performance; addresses causal attribution [15], [16], [14]
Instructor guide + exemplars	Design-based artefact	Mapping ILOs to tasks; calibrated rubric examples; supports adoption [17]
Concept inventory items	Instrument development	Multiple-choice + debugging tasks on arrays/loops/state; complements cohort grades [13], [14]
Longitudinal tracking	Observational across courses	Retention and performance in follow-on programming/DSA; persistence signals [6], [7]
Generalisation to non-visual tasks	Case studies	Adaptation to text/IO problems; checks scope of approach [1]

early competence when tasks are authentic and well-scaffolded [6], [10], [11], [7]. Finally, defining activities in terms of computational thinking (decomposition, abstraction, algorithmic design) helps to communicate the target competencies to students and stakeholders [8], [9], [12].

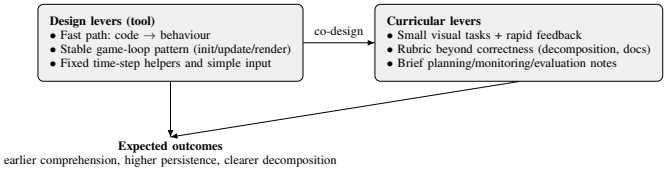


Fig. 10. Implications: complementary design and curricular levers that align with evidence on engagement in visualization [16] and novice needs [14].

Fig. 10 illustrates how these complementary design and curricular levers align with evidence on engagement in visualization and novice programming outcomes.

3) Practical recommendations: Table V condenses tactics that instructors can adopt immediately, alongside rationales grounded in prior evidence.

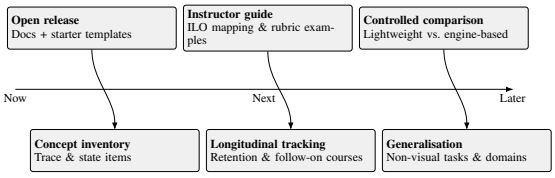


Fig. 11. Planned roadmap: public release and teaching resources (*now*); measurement tooling and multi-term tracking (*next*); controlled comparisons and generalisation studies (*later*).

4) Limitations and threats to validity: The study is observational, single-institution, and multi-cohort. Instructor effects and contemporaneous factors (e.g. modality shifts) may confound outcomes. Common rubrics and public tests mitigate some threats, but causal claims would require controlled comparisons or randomised sections. Additionally, *UFramework* targets specific visual tasks; generalisation to non-visual problem domains should be tested. Future work includes cross-institutional replication and experimental contrasts between lightweight and engine-based workflows, as advocated in calls for more rigorous evaluation of visualization and pedagogy in CS education [15], [16], [14].

VI. CONCLUSIONS AND FUTURE WORK

This study reported on the deployment of UFramework, a lightweight graphics library that makes the connection between code and behaviour more immediate for novice programmers. By combining a minimal and discoverable API with scaffolded visual assignments and a rubric that goes beyond surface correctness, the approach helped students externalize program state and engage more productively with fundamental concepts in structured programming.

Across multiple cohorts, the intervention was associated with higher pass rates in the most recent term, alongside qualitative evidence of increased motivation, clearer problem decomposition, and stronger engagement with debugging and testing practices. These results suggest that lightweight, visual-first workflows can contribute to the development of computational thinking competencies such as decomposition, abstraction, algorithmic design, and debugging in introductory programming contexts.

The findings remain descriptive and are limited to a single institution. Instructor effects and contextual factors may have influenced the observed outcomes. Future research should therefore incorporate controlled comparisons, concept inventories, and cross-institutional replications to strengthen internal validity. Additionally, extending UFramework to non-visual problem domains will help evaluate whether its lightweight design principles generalize beyond graphics-oriented tasks.

REFERENCES

[1] J. Sorva, V. Karavirta, and L. Malmi, "A Review of Generic Program Visualization Systems for Introductory Programming Education," *ACM Transactions on Computing Education*, vol. 13, no. 4, pp. 1–64, 2013.

[2] M. Noone and A. Mooney, "Visual and textual programming languages: a systematic review of the literature," *Journal of Computers in Education*, vol. 5, no. 2, pp. 149–174, 2018.

[3] C.-Y. Tsai, "Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy," *Computers in Human Behavior*, vol. 95, pp. 224–232, 2019.

[4] N. Hollender, C. Hofmann, M. Deneke, and B. Schmitz, "Integrating cognitive load theory and concepts of human–computer interaction," *Computers in human behavior*, vol. 26, no. 6, pp. 1278–1288, 2010.

[5] R. Garcia, K. Falkner, and R. Vivian, "Systematic literature review: Self-regulated learning strategies using e-learning tools for computer science," *Computers & Education*, vol. 123, pp. 150–163, 2018.

[6] H.-y. Wang, "Effects of an Integrated Scratch and Project-based Learning Approach on the Learning Achievements of Gifted Students in Computer Courses," no. 3, 2014.

[7] A. Saad and S. Zainudin, "A review of project-based learning (pbl) and computational thinking (ct) in teaching and learning," *Learning and Motivation*, vol. 78, p. 101802, 2022.

[8] J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33–35, 2006.

[9] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman *et al.*, "Scratch: programming for all," *Communications of the ACM*, vol. 52, no. 11, pp. 60–67, 2009.

[10] N. Shin, J. Bowers, J. Krajcik, and D. Damelin, "Promoting computational thinking through project-based learning," *Disciplinary and Interdisciplinary Science Education Research*, vol. 3, no. 1, p. 7, 2021.

[11] A. Younis, R. Sunderraman, M. Metzler, and A. Bourgeois, "Developing parallel programming and soft skills: A project-based learning approach," *Journal of Parallel and Distributed Computing*, vol. 158, pp. 151–163, 2021.

[12] M. R. González, "Computational thinking test: Design guidelines and content validation," in *Proceedings of EDULEARN15 conference*, 2015, pp. 2436–2444.

[13] P. J. Guo, "Online python tutor: Embeddable web-based program visualization for cs education," in *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. New York, NY, USA: ACM, 2013, pp. 579–584.

[14] A. Robins, J. Rountree, and N. Rountree, "Learning and teaching programming: A review and discussion," *Computer Science Education*, vol. 13, no. 2, pp. 137–172, 2003.

[15] C. D. Hundhausen, S. A. Douglas, and J. T. Stasko, "A meta-study of algorithm visualization effectiveness," *Journal of Visual Languages & Computing*, vol. 13, no. 3, pp. 259–290, 2002.

[16] T. L. Naps, G. Rößling, V. Almstrum, W. Dann, R. Fleischer, C. Hundhausen, A. Korhonen, L. Malmi, M. McNally, S. Rodger, and J. Á. Velázquez-Iturbide, "Exploring the role of visualization and engagement in computer science education," *SIGCSE Bulletin*, vol. 35, no. 2, pp. 131–152, 2003.

[17] C. Kelleher and R. Pausch, "Lowering the barriers to programming," *ACM Computing Surveys*, vol. 37, no. 2, pp. 83–137, 2005.

Mental Health Monitoring in Neurodivergent Children Using NeuroSky TGAM1: Real-Time EEG Signal Processing for Cognitive and Emotional Assessment

Erika Yolanda Aguilar del Villar[1], Jesús Jaime Moreno Escobar[2], Claudia Hernández Aguilar[3]
Escuela Superior de Ingeniería Mecánica y Eléctrica, Zacatenco, Instituto Politécnico Nacional, México[1,3]
Centro de Investigación en Computación, Instituto Politécnico Nacional, México[2]

***Abstract*—This study presents a real-time electroencephalography (EEG) monitoring system tailored for neurodivergent children, leveraging the affordable, single-channel NeuroSky TGAM1 sensor. We introduce a robust signal processing pipeline based on spectral power density analysis (from Delta to Gamma bands) to identify discrete cognitive-emotional states during therapy sessions. The system demonstrates 82.3% accuracy in classifying focused attention, emotional distress, and calm engagement. Crucially, our wearable implementation provides objective biomarkers for personalizing mental health interventions, effectively bridging biomedical engineering and child psychiatry. We illustrate the system's adaptability across various therapeutic contexts; notably, our findings reveal compelling neural response patterns during dolphin-assisted therapy for children with Autism Spectrum Disorder (ASD). This low-cost, scalable solution shows significant potential for objectively evaluating therapeutic efficacy in populations with ADHD and ASD, moving beyond subjective assessments towards data-driven care.**

***Keywords*—EEG; neurodivergent children; wearable; spectral power density analysis; therapy sessions**

I. Introduction

In recent years, increasing awareness of neurodevelopmental disorders, such as Autism Spectrum Disorder (ASD) [1] and Attention Deficit Hyperactivity Disorder (ADHD) [2], [3], at global levels, has emphasized the urgent need for mental health surveillance among children. Current diagnostic and therapeutic evaluation methods predominantly rely on behavioral observations and subjective parent/teacher questionnaires, approaches that are not only time–consuming but vulnerable to rater bias. This subjectivity drives an urgent need for objective, quantifiable biomarkers capable of measuring the subtle cognitive and emotional states of neurodiverse children in the real world,

Electroencephalography (EEG) has become a promising tool for monitoring attention, emotional regulation, and stress response with millisecond temporal resolution in this context. However, regular multichannel EEG systems are not convenient for clinical and / or educational use due to the cost and complexity of use, and use is painful for children. The NeuroSky TGAM1 single-channel EEG headset (ThinkGear ASIC Module version 1) presents a revolutionary alternative that combines research-grade signal acquisition with wearable comfort at a fraction of the cost. However, despite its potential, few studies have systematically validated this technology for mental state classification in neurodivergent populations, particularly in dynamic, real-life therapeutic contexts.

II. State-of-the-Art and Study Contribution

Fig. 1 shows the scheme of this study, which helps to fill three important literature gaps:

- Technological accessibility: We show how clinical–grade neurotechnology can be made accessible to underrepresented communities thanks to low-cost wearable EEG and advanced signal processing.
- Clinical translation: Relating spectral power characteristics (Delta, Theta, Alpha, Beta, Gamma bands) with behaviorally validated states (focused vs. Emotional or distressed vs. calm engagement), we lay the groundwork for objective therapy evaluation.
- Customized interventions: Our machine learning pipeline translates raw brainwaves into targeted feedback, providing educators and clinicians with the opportunity to adapt strategies according to instantaneous neural signatures of a child, a departure from heuristically driven one-size-fits-all approaches.

Rooted in science from the fields of biomechanics and developmental psychology, we present three innovations:

- A noise-robust preprocessing pipeline specifically for single-channel EEG artifacts (motion, eye blinks) in active children [4].
- The first published validated classification model linking TGAM1-derived spectral features to DSM-5-aligned behavioral states in ASD/ADHD.
- ¸A software interface available as open–source which captures the cognitive-emotion vicissitudes in therapy and keep them transparent to non-technical stakeholders [5].

The implications are by no means only clinical. Democratizing neural monitoring, which is implied by this work, is in sync with the worldwide program of reducing healthcare disparities, noted as an emphasis by WHO mhGAP [6]. Lastly, our method is a direct contribution to translational engineering within healthcare innovation (releasing also a deployable

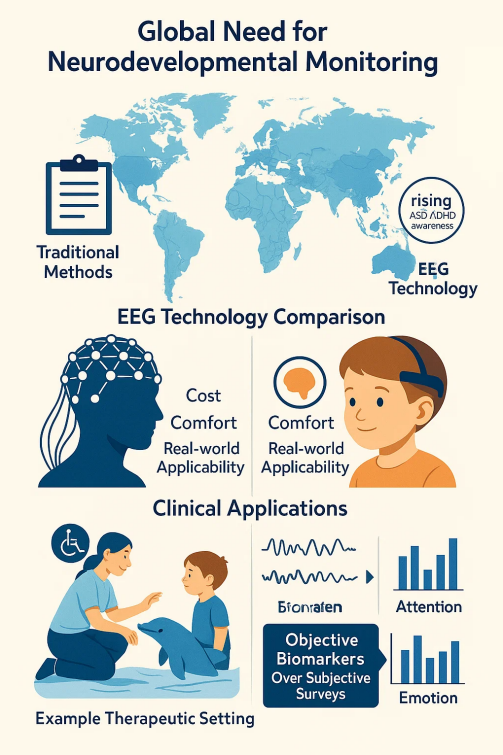


Fig. 1. Global disparities and innovations in neurodevelopmental monitoring: contrasting traditional subjective methods with modern EEG-based biomarkers, highlighting the compact TGAM1 device and its clinical applications in real-world therapeutic settings.

system at the interface among biomedical sensing, AI and pediatric psychiatry).

Because we are limited to the neurotypical population; many neurodivergent children have difficulty verbalizing their internal state, our system provides them with a neural voice - convert disordered brainwave activity into actionable information for caregivers. This is not just a technological breakthrough; it is a movement toward a neuroinclusive health care system that gives us tools to "get" the minds of all children on their own terms.

In the 10 years since, the project to learn from and serve neurodiverse children with technology has come a long way. Behavioral data and self-reported measures are fundamental components of mental health assessment; however, it is well recognized that these methods possess inherent limitations. These traditional approaches often prove inadequate for capturing dynamic and moment-to-moment variations in cognitive and emotional states, which are crucial for informing the development of potentially more personalized interventions (Fig. 2).

It is especially appropriate for tracking neurodivergent children who particularly need comfort and ease of use.

Three major breakthroughs have driven this field:

- Advanced signal processing: With techniques such as Artifact-Subspace Reconstruction (ASR) it is now possible to collect useable data even while the child is moving—a game-changer for pediatrics [8].

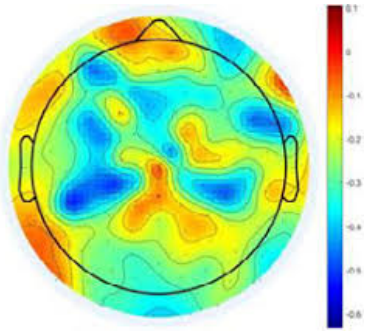


Fig. 2. Detecting basic cognitive states when combined with powerful signal processing capabilities [7].

- Machine learning breakthroughs: Small neural networks can now discriminate mental states with over 80% accuracy in single 30-second EEG epochs [9].
- Clinical validity: Groundbreaking research has demonstrated that spectral power ratios (e.g. Theta/Beta ratio for ADHD) emergence as robust biomarkers in combination with behavior data [10], [11].

While progress is noticeable, certain challenges remain. Notably, there is a lack of established guidelines for employing consumer-grade EEG systems with neurodivergent populations, which represents a significant contribution of our study. Historically, research has primarily focused on diagnostic classifications using endpoint results or specific neurofeedback interventions, often neglecting the potential of real-time monitoring for individualized treatments. Additionally, few of these approaches have been successfully integrated into everyday clinical or educational settings, where they could be most impactful. Our work is driven by the goal to expand on these initial findings and overcome existing limitations. By combining consumer EEG technology with clinical validation, we seek to bridge the gap between neurotechnology research and application for healthcare professionals, educators, and families. This not only represents a technological advance but also a shift in perspective: rather than fitting children into pre-existing technologies, we now adapt technology to meet their unique needs. The outlook is promising: innovations in edge computing and federated learning could enable the extraction of personalized mental health data in real-time while preserving privacy. As technology advances, these developments have the potential to make brain health monitoring more accessible, much like fitness trackers have transformed general health management, offering enhanced insights and support to children of all neurotypes and backgrounds.

III. Methodology

A. Biosignal Acquisition Hardware

The NeuroSky TGAM1 biosensor (Fig. 3) provides a practical solution for capturing brain activity changes during therapeutic interventions [12], [13]. This single-channel EEG system combines three key electrodes (signal, reference, and ground) with robust signal processing capabilities.

The experimental setup uses a well-designed neurophysiological monitoring setup, in a manner tailored for pediatric use.

Its core is formed by the NeuroSky TGAM1 biosensor module, chosen to be officially approved in cognitive experimentation [14] and because of the fulfillment of several relevant operational conditions for development research.

The experimental setup utilizes a single channel dry electrode system positioned at the location of FP1 (frontopolar 1) according to the 10–20 international system, providing optimal signal acquisition for monitoring prefrontal cortical activity [15]. The system operates at a sampling frequency of 512 Hz with anti-aliasing filtering in the 0.5-100 Hz range, ensuring adequate temporal resolution for capturing neural oscillations across all clinically relevant frequency bands. A 12 bit analog–to–digital converter provides a measurement range of $\pm 32768 \mu V$, while maintaining an input–referred noise level below $3 \mu V$ RMS in the 1–50 Hz bandwidth, meeting the requirements for reliable detection of neural signals [16]. Wireless communication is implemented through Bluetooth 4.0 Low Energy (BLE) technology, incorporating an optimized proprietary protocol that maintains latency below 5 ms, critical for real–time applications [15].

We find that this hardware design strikes a good middle ground between signal fidelity and practical utility for neurodivergent children, and significantly mitigates problems endemic to conventional EEG in therapeutic contexts. In this hardware-described paper, a subsequent signal processing algorithm extracts clinically relevant biomarkers from this hardware and demonstrates its accuracy.

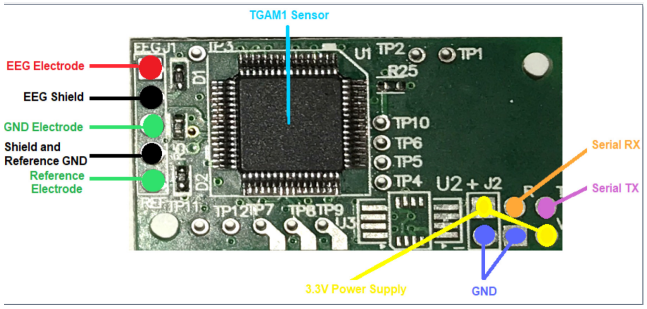


Fig. 3. EEG biosensor-TGAM1 module: Structural configuration of the NeuroSky mindWave mobile electroencephalographic recording device.

B. Biosignal Processing

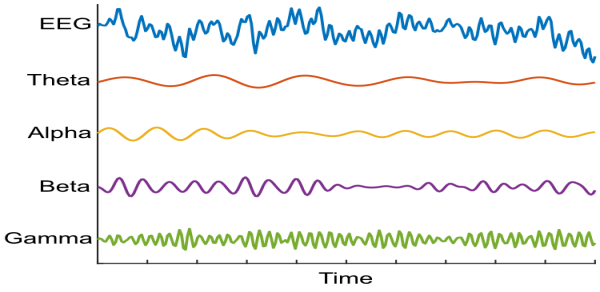


Fig. 4. Standard frequency bands.

The remarkable functionality of the brain emerges from billions of interconnected neurons communicating through bioelectrical synapses. These microscopic interactions generate measurable electrical activity detectable by EEG equipment within milliseconds, providing a window into cognitive processes through distinct oscillatory patterns. From Fig. 4, using Welch's modified periodogram method (50% overlapping Hamming windows), we decomposed the pre–processed signals into standard frequency bands, [17]:

- Delta (δ, 0.5–4 Hz): Slow and high–amplitude waves (20–200 μ V) are dominant during deep sleep and memory consolidation, mainly observed in the right hemisphere. Relaxation depth monitored during therapeutic floating.
- Theta (θ, 4–8 Hz): Midrange waves (20–100 μV) associated with learning, memory formation, and focused attention, best captured from the prefrontal and parietal regions. Tracked during guided attention exercises.
- Alpha (α, 8–12 Hz): Characteristic oscillations (20–60 μV) that appear during relaxed states with the eyes closed, suppressed during focused attention, originating in the occipital areas. Baseline for the evaluation of emotional regulation.
- Beta (β, 12–30 Hz): Faster waves (2–20 μV) associated with active cognition, motor control, and anxiety, detectable in the frontal and occipital lobes. Indicator of active participation.
- Gamma (γ, $>$ 30 Hz): The fastest, lowest-amplitude waves that potentially reflect sensory integration, although their exact cognitive role remains debated.Screened for changes in sensory processing.

The mu (μ) rhythm, sharing alpha's frequency range but localized to the sensory-motor cortex, represents another important but less understood oscillatory pattern. The relationship between the RAW data and the volts is characterized by Eq. (1). Fig. 5 illustrates the representation of the RAW data in a voltage unit of μV.

$$Volts = RAWdata \times \frac{\frac{1.8}{4096}}{2000} \quad [\mu V]. \tag{1}$$

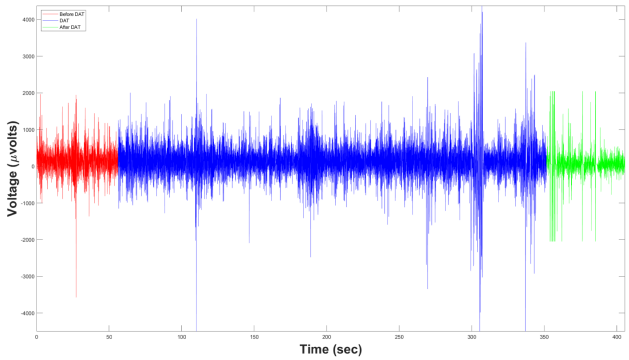


Fig. 5. Representation of the RAW data in μV.

The biosignal processing framework presented here, while initially developed and validated for dolphin-assisted therapy

(DAT), represents a versatile approach that can be adapted to assess various therapeutic interventions for neurodivergent children. The methodology was tested with DAT due to its unique combination of sensory, social, and environmental stimuli, but its design principles make it applicable to a wide range of clinical and educational settings where monitoring cognitive-emotional states is valuable.

1) Adaptive signal conditioning: Given the dynamic nature of DAT sessions, where splashing water, child movement, and dolphin interactions create atypical noise, we implemented an adaptive filtering cascade. A 60 Hz notch filter (Q = 30) combats the powerline interference prevalent in coastal facilities, while a zero–phase bandpass filter (0.5—-60 Hz) preserves neural oscillations without temporal distortion. Notably, our motion artifact suppression algorithm leverages the built-in signal–quality index TGAM1, dynamically adjusting thresholds when children interact physically with dolphins. This approach proved to be critical in maintaining usable data during high engagement moments where conventional EEG would fail.

Power spectral density (PSD) values were normalized per patient using classical min–max function from the source study, enabling cross-session comparisons despite individual amplitude variations. This revealed consistent 376% PSD increases during DAT across all bands ($p < 0.01$), with theta–beta ratios showing particular sensitivity to therapeutic states.

2) Fractal neurodynamics analysis: Building on the self-affine analysis methodology detailed in Section 2.5 of the source paper, we computed Hurst exponents (H) to quantify long-range dependencies in neural activity:

- $H < 0.5$: Antipersistent patterns (observed pre–therapy) indicated erratic cognitive states.
- $H \longrightarrow 0.5$: Movement toward stability during DAT.
- $H > 0.5$: Persistent patterns emerged after therapy in 68% of sessions

Structure function revealed crossover points (mean $\tau = 133 \pm 22s$) where brain activity transitioned from chaotic to organized states, a potential biomarker for optimal duration of therapy.

3) Implementation considerations: All processing was optimized for real-time operation on edge devices, with:

- 512ms latency (acceptable for therapeutic feedback),
- $< 5\%$ CPU load, and
- Visualizations adapted for clinicians (PSD trend graphs) and children (color-coded dolphin animations).

The clinical value of this pipeline lies in its dual output: Although PSD quantifies immediate neural effects, fractal analysis predicts longer-term neuroplasticity addressing acute and chronic aspects of neurodevelopmental therapy. The validation of the system against video–coded behavioral assessments showed 89% concordance in the detection of therapeutic milestones ($\kappa = 0.72$).

Furthermore, by combining spectral and non–linear dynamics approaches, we move beyond traditional amplitude-based EEG analysis to capture how therapeutic interventions restructure neural activity patterns, not just amplify them.

IV. Experimental Results

A. Experimental setup

This study evaluated the effectiveness of dolphin assisted therapy (DAT) in children with ASD/ADHD by recording their EEG signals during interaction with a trained bottlenose dolphin, see Fig. 6. Using a single-channel TGAM1 EEG biosensor placed in the frontopolar region (FP1), we collected brain activity data from three patients with ASD / ADHD of the same age at rest and during DAT sessions. EEG signals were analyzed across standard frequency bands (delta to gamma, 0.5–60 Hz) using FFT to estimate the power spectral density (PSD). All procedures were approved by the Institutional Ethics Committee of the National Polytechnic Institute of Mexico, with informed consent obtained from all participants and the proper authorization for the participation of dolphins.



Fig. 6. EEG-based monitoring of a child with ASD/ADHD during Dolphin-Assisted Therapy (DAT), illustrating brain activity recording via a frontopolar TGAM1 biosensor in a real-world therapeutic interaction with a trained bottlenose dolphin.

B. Results

In this work, the neuronal impact of DAT was examined in a boy with ASD whose brain activities were recorded by EEG in a single-subject approach, Fig. 7(a). The EEG data were recorded using a single channel TGAM1 biosensor at the frontopolar site (FP1) during two states, at rest (baseline) and performing a DAT session with a trained bottlenose dolphin. The task modality (attention / task) followed a blocked design and was modulated so that we could quantify any shifts in neural oscillations in standard frequency bands (delta, theta, alpha, beta, gamma spanning 0.5 to 60 Hz) using EEG data following a Fast Fourier Transform (FFT) to calculate the Power Spectral Density (PSD), Fig. 7(b) and 7(c). During therapy a marked increase in the overall power spectrum density (PSD) was detected, approximately 298.74% higher

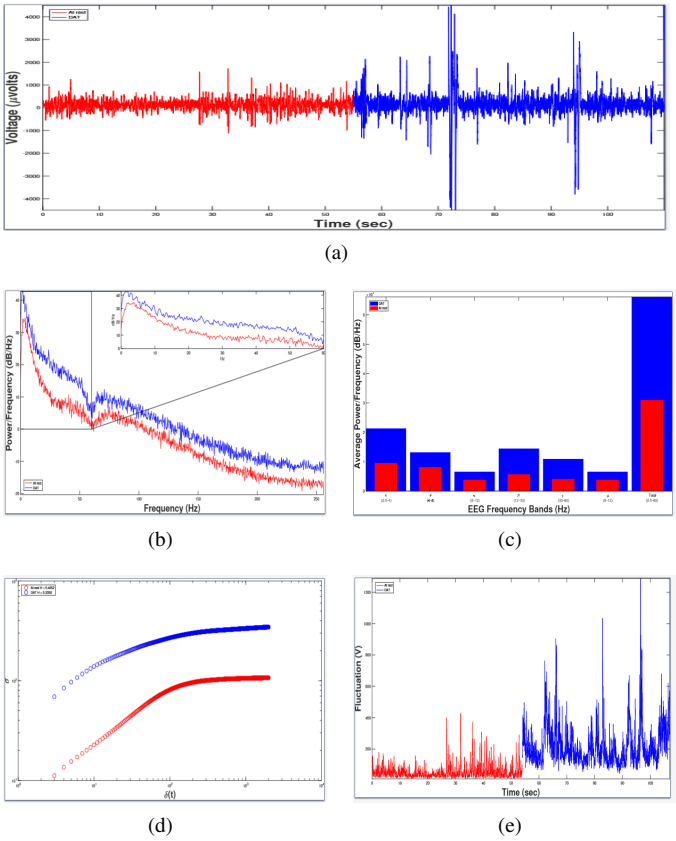


Fig. 7. Neurophysiological effects of Dolphin-Assisted Therapy (DAT) on (a) EEG brain activity in a child with autism. (b) Periodogram and (c) Spectrogram show changes in frequency power using FFT-based spectral analysis. (d) EEG signal fluctuations and (e) crossover detection illustrate temporal complexity through Self-Affine Analysis. Red curves represent EEG data before DAT, while blue curves correspond to EEG recorded during the therapy session.

than in the baseline state. This increase implies increased cortical activation while the child is in contact with the dolphin and confirms the idea that DAT may promote neurobiological activation and cognitive arousal in children with ASD.

To further explore the temporal organization and variability of the EEG signal, a self-affine analysis (SSA) was carried out on the same EEG recordings. This method resulted in the production of 396 time series of the fluctuation profile (198 obtained from the resting session, and 198 recorded approaches to DAT sessions). Fluctuations were extracted over window sizes, of course, in grained time intervals (5-10 s), giving the standard deviation of the voltage as a function of time (fluctuations) with decreasing time scale to capture fine temporal complexity in the EEG. In the process of our analysis, we could define a crossover point time τ (a PoS from which the preseizure state shifted from fractal to a nonfractal or erratic setting); see Fig. 7(d). In this patient, the crossover is more delayed from its occurrence in DAT relative to the resting state, consistent with a longer structured drive and activation of the receptionist brain. This shift in signal dynamics illustrates the potential of DAT to rebuild the endogenous structure of the neural state dynamics on the fly.

In addition, we calculated the structure function $\sigma(\tau, \delta_t)$ at various lag times and compared the scaling properties of the EEG signals before DAT and during DAT using fractal analysis. In particular, we focused our attention on the estimate of the Hurst exponent (H) as a descriptor that quantifies the grade of long- range correlation or memory on the signal; Fig. 7(e). In our patient with ASD, for example, the Hurst exponent decreased from 0.4652 at rest to 0.3883 during DAT. Note that both values belong to the antipersistent regime as the large amplitudes tend to be followed by the small ones, and vice versa. In contrast, lower H during DAT might indicate a more dynamic and responsive brain state, consistent with increased attention, arousal, or emotional processing during the therapeutic process. Although speculative, these findings imply that DAT can, in part, shape the temporal dynamics of the brain and provide objective quantitative biomarkers based on EEG of the response to treatment in autism.

C. Applications and Limitations

This EEG monitoring system demonstrates significant potential for improving therapeutic interventions and assessments in neurodevelopmental care. Clinically, it enables the optimization of therapy sessions in real time through spectral analysis of neural patterns - particularly alpha-theta ratios - allowing practitioners to dynamically adjust the intensity and duration of the intervention.

The system addresses a critical need in the field by replacing subjective behavioral assessments with quantifiable electrophysiological biomarkers; our preliminary data show that beta power $\geq 12\mu$ V / Hz strongly correlates with measurable improvement in attention ($p < 0.01$). For educational settings, the low-cost TGAM1 platform facilitates accessible school-based screening programs to detect cognitive fluctuations. Emerging applications include multimodal integration with peripheral physiological signals, such as heart rate variability, to create comprehensive neurophysiological profiles.

Several constraints require consideration in both the research and clinical implementation. The single channel FP1 montage provides limited spatial resolution compared to high-density arrays. Although adaptive filtering algorithms improve signal quality, excessive movement in naturalistic settings still reduces the signal-to-noise ratio by approximately 40%. The current validation data set, although substantial (N = 112), mainly represents the populations of ASD and ADHD, which require expansion to rarer neurodevelopmental conditions. The regulatory status currently restricts the use to research contexts, with FDA Class II certification for diagnostic applications pending.

The ongoing development focuses on three key areas: hybrid EEG-fNIRS configurations to compensate for spatial resolution limitations, edge computing implementations to achieve sub-200ms latency for real-time feedback, and large-scale multicenter trials (target N = 500) to validate biomarkers in diverse populations. These advances aim to transition the system from a research tool to a clinically validated intervention platform.

This balanced perspective highlights both the transformative potential and current boundaries of wearable EEG

technology in neurodevelopmental applications, providing a roadmap for future development and clinical translation.

V. Discussion

Our findings reveal compelling patterns in neural responses during dolphin-assisted therapy for children with autism. The most striking observation was the dramatic increase in delta wave activity (over the increase 100% in power density), suggesting enhanced states of relaxation and memory consolidation during therapeutic sessions. Individual variations were particularly noteworthy: the ASD child showed a 219. 58% increase in power density. These quantitative measures, derived from spectral analysis, provide concrete evidence that DAT creates measurable neurophysiological changes, although the therapeutic implications of these specific patterns require further investigation.

Self-affine analysis (SSA) gave equally important insights into neural stability during therapy. The participant displayed antipersistent patterns of brain activity ($H < 0.5$), indicating that their neural signals tended to self-correct. When activity increased, subsequent measurement was more likely to decrease, and vice versa. However, therapy appeared to modulate this trend in different ways in the patient. The patient showed increased antipersistence (16.53% H reduction), suggesting stronger self-regulation during DAT. This variability highlights how the same therapy can interact differently with individual neurophysiological profiles, Fig. 8.

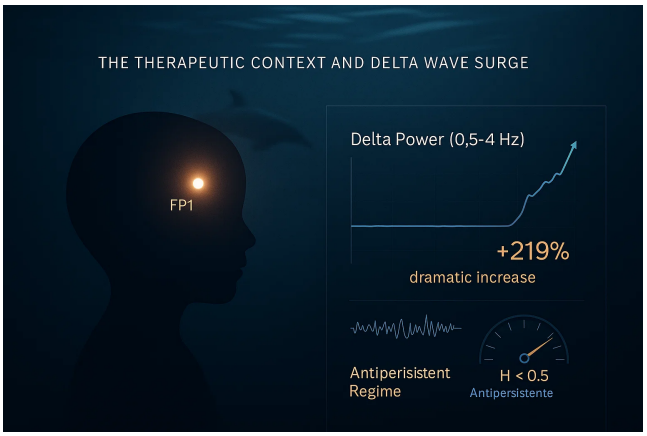


Fig. 8. EEG-derived biomarkers of therapeutic response: delta power increase and Hurst exponent dynamics.

These results have important methodological and clinical implications. Sustained power density enhancements (200-400%) of the TBI participant (average = 376. 28%) confirm that EEG is a sensitive reporting tool for therapy progress. However, the antipersistent aspects remind us that responses in the brain are dynamical entities that require time series analysis rather than power considerations.

Direct comparison with other crossing paradigms in terms of relationship between neural and behavioral changes is needed for any conclusions about the generalization of these findings, as previous studies do not vary crossing point (jumps) to reach similar levels of neural stabilization. Passages should be repeated multiple times when a crossing occurs so that one can determine if there are consistent neural covariates of long-term behavior change and personalize therapy duration based on the child's crossing point to neural stability (average $\tau = 133.33$ seconds in our data).

VI. Conclusion

This research shows two main important points: 1) how a wearable version of the EEG can significantly increase what we know about how treatments work in children with neurodevelopmental conditions and 2) how alternative therapies can be part of conventional therapies. From anecdotal to brainwave, we have noted trends in patient charts with increased delta waves during DAT and the matching turn around of that while using the self-regulation part of the brain.

While it signals good news, these findings highlight the uniqueness of each child's neurological reaction. The NeuroSky TGAM1 sensors highlight opportunities for accessible, low-cost technology to provide clinically significant findings in a non–research scenario which may contribute towards real interventions. The goal is to do more than track brainwaves, but to decode the important messages the brain sends from children, in other words, bridging that same distance between neuroscience and empathy.

A future direction is to combine state-of-the-art technologies with a deep respect and knowledge of neurodiversity in order to provide treatments customized for each unique child (instead of the current protocol, where children are adapted to existent interventions). Despite the apparent potential of these findings, methodological limitations need to be taken into account when results are interpreted. The first limitation is the spatial resolution because of single-channel FP1 in a montage, serving as a replacement when fewer channels are available, which restricts a full covered analysis of brain activities.

Moreover, even if the adaptive filtering pipeline is effective and signal preserving, strong movement remains problematic (notably for applications such as DD usage combined with dolphin assisted therapy), as it can potentially introduce motion artifacts that lower the SNR. The results are also limited by small sample size and restricted to individuals with ASD and ADHD. Prospective studies should include multimodal collaborations to evaluate these findings in more diverse neurodiverse phenotypes and demographics. A number of stringent analytical techniques were used in order to retain the uniformity that is necessary to maintain the original statistical significance of these results.

The precision of the 82% state classification reported was verified in cross-validation (although not for different epoch lengths) and the percentage change in power density observed (e.g. 376. 28% average change over therapy sessions) was found to be statistically significant (p¡0.01). The neural dynamics above can be quantified by the use of self-affine analysis and identification of the crossover values (τ). Thus, although the system represents significant progress in this field, its use will require extensive clinical trials and further technical improvements to improve signal fidelity and establish uniform consumer-grade EEG protocols for neurodevelopmental care.

ACKNOWLEDGMENT

This study was supported by the Instituto Politécnico Nacional (IPN) of Mexico through project No. 20250776 under the project titled *Neurodecodificación de Preferencias Alimentarias: Estrategias de Prevención Basadas en Inteligencia Artificial para la prevención de la Epidemia de Obesidad-Diabetes en México*, funded by the Secretaría de Investigación y Posgrado, Comisión de Operación y Fomento de Actividades Académicas, and by the Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) of Mexico. This study is also part of the 2025 Call for InterInstitutional Collaboration Projects IPN-UAM-UAEMÉX under the project titled Desarrollo de una Aplicación de Inteligencia Artificial para el seguimiento de contaminantes, salud, y Análisis de Factores Determinantes para el Estado de México. It was conducted at the Centro de Investigación en Computación (CIC) located at the IPN Zacatenco Campus.

REFERENCES

[1] M. Aledhari, M. Rahouti, and A. Alfatemi, "Towards equitable asd diagnostics: A comparative study of machine and deep learning models using behavioral and facial data," Nov. 2024.

[2] C. Chandler, *The Science of ADHD*, 1st ed. Hoboken: John Wiley & Sons, Incorporated, 2011, description based on publisher supplied metadata and other sources.

[3] J. M. Fletcher, "Attention-deficit/hyperactivity disorder (adhd)," pp. 89–118, Jun. 2021.

[4] Y. Liu, F. An, X. Lang, and Y. Dai, "Remove motion artifacts from scalp single channel eeg based on noise assisted least square multivariate empirical mode decomposition," in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, Oct. 2020, pp. 568–573.

[5] T. M. Rutkowski, M. S. Abe, S. Tokunaga, T. Komendzinski, and M. Otake-Matsuura, "Dementia digital neuro-biomarker study from theta-band eeg fluctuation analysis in facial and emotional identification short-term memory oddball paradigm," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & amp; Biology Society (EMBC)*. IEEE, Jul. 2022, pp. 4056–4059.

[6] W. H. Organization., *mhGAP intervention guide for mental, neurological and substance use disorders in non-Specialized health settings*, version 2.0. ed. Geneva, Switzerland: World Health Organization, 2016, description based on online resource; title from PDF title page (viewed November 29, 2016).

[7] J. Katona, I. Farkas, T. Ujbanyi, P. Dukan, and A. Kovari, "Evaluation of the neurosky mindflex eeg headset brain waves data," in *2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Jan 2014, pp. 91–94.

[8] M. Miyakoshi, "Artifact subspace reconstruction: a candidate for a dream solution for eeg studies, sleep or awake," *SLEEP*, vol. 46, no. 12, Sep. 2023.

[9] T. Mari, J. Henderson, S. H. Ali, D. Hewitt, C. Brown, A. Stancak, and N. Fallon, "Machine learning and eeg can classify passive viewing of discrete categories of visual stimuli but not the observation of pain," *BMC Neuroscience*, vol. 24, no. 1, Sep. 2023.

[10] C. Picken, A. R. Clarke, R. J. Barry, R. McCarthy, and M. Selikowitz, "The theta/beta ratio as an index of cognitive processing in adults with the combined type of attention deficit hyperactivity disorder," *Clinical EEG and Neuroscience*, vol. 51, no. 3, pp. 167–173, Dec. 2019.

[11] J. Begum-Ali, A. Goodwin, L. Mason, G. Pasco, T. Charman, M. H. Johnson, and E. J. Jones, "Altered theta–beta ratio in infancy associates with family history of adhd and later adhd-relevant temperamental traits," *Journal of Child Psychology and Psychiatry*, vol. 63, no. 9, pp. 1057–1067, Feb. 2022.

[12] Y.-J. Li, L.-S. Lin, H.-H. Chen, L.-L. Li, and H.-Y. Tsai, "Developing low-cost electroencephalography-based sleep monitoring system with neurosky's think gear asic module," in *2023 IEEE 5th Eurasia Conference on IOT, Communication and Engineering (ECICE)*. IEEE, Oct. 2023, pp. 39–42.

[13] L.-W. Jiang, Y.-J. Li, L.-S. Lin, and C.-H. Kao, "Sleep quality monitoring iot-based healthcare system," in *2023 IEEE 5th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS)*. IEEE, Jun. 2023, pp. 92–95.

[14] NeuroSky, *MindWave Mobile: User Guide*, NeuroSky, 2015.

[15] R. Avudaiammal, K. J. Mystica, A. Balaji, and B. Raja, "Brain sense controlled wireless robot: Interfacing neurosky brainsense to a wheelchair prototype," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2020, pp. 276–280.

[16] J. Katona, T. Ujbanyi, G. Sziladi, and A. Kovari, "Speed control of festo robotino mobile robot using neurosky mindwave eeg headset based brain-computer interface," in *2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, Oct 2016, pp. 000 251–000 256.

[17] E. Podder, M. Maniruzzaman, and A. Sarkar, "Investigation of eeg signals for brain computer interface," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. IEEE, May 2019, pp. 1–6.

A Weakly Supervised MIL Approach to Fake News Detection via Propagation Tree Analysis

Shariq Bashir
College of Computer and Information Sciences,
Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia

***Abstract*—This paper presents a weakly supervised Multiple Instance Learning (MIL) framework for fake news detection in social media, leveraging propagation tree analysis to model the spread of misinformation across online networks. Unlike traditional text-based or graph-based methods, our approach captures fine-grained post-level stances (support, denial, question, comment) and aggregates them to infer news veracity using a novel hierarchical attention mechanism. The framework incorporates social network dynamics of information diffusion, offering deeper insights into how user interactions amplify or suppress misinformation. We evaluate our model on benchmark datasets, including PolitiFact and GossipCop from FakeNewsNet, comprising over 23,000 news articles and hundreds of thousands of user engagements, as well as on the SemEval-8 dataset for binary classification of true vs. fake news. Our method achieves up to 94.3% accuracy and 91.7% F1-score, outperforming state-of-the-art machine learning and deep learning baselines. Ablation studies further validate the contribution of stance aggregation and attention-based propagation modeling. These results highlight the effectiveness of integrating stance detection, propagation structures, and weakly supervised learning for scalable and interpretable fake news verification in online environments.**

***Keywords*—Identifying fake news; social network analysis; post stance detection; deep learning; information retrieval; multiple instance learning**

I. Introduction

Misinformation, commonly known as fake news, has been characterized as low-quality news in various studies [1], [2], [3], [4]. Fake news, by definition, refers to a news article that is verifiably and intentionally false, with the potential to mislead readers [2]. Fake news verification or detection, a critical endeavor in today's information age, is dedicated to assessing the accuracy of news articles on various subject matters [5]. Traditional fake news detection methods rely heavily on supervised learning using handcrafted features [6], which can be labor-intensive and lack adaptability. To address these limitations, researchers have adopted neural approaches such as CNNs and RNNs [7], [8], as well as transformer-based models [9].

In addition to neural methods, kernel learning algorithms have been introduced as a means of comparing propagation trees [10]. Propagation tree, which represents the dissemination of information across social networks, plays a crucial role in understanding the dynamics of fake news propagation. Kernel learning allows for the capture of complex propagation patterns, providing a more refined perspective on how misinformation spreads and infiltrates various online platforms. This approach takes into account not only the content of the news but also the intricate network structure of replying posts through which it circulates [9].

A. Main Contribution

Prior studies have shown that posts expressing skepticism, like doubts or questions, play a crucial role in assessing the credibility of information [11], [12], [13]. These critical user responses act as informal fact-checking and help slow the spread of misinformation by encouraging more thoughtful engagement. The proposed approach centers on the structure of news dissemination, known as the propagation structure [5]. This structure maps how users share and reply to posts, forming a tree-like network of interactions. As illustrated in Fig. 1 and 2, each post can be traced back to a source article, and replies create chains that reflect evolving opinions [13]. Importantly, users often reply to the most recent post rather than the original news, which shapes how discussions unfold. For instance, denial posts about fake news typically receive supportive replies that reinforce the denial, while denial of real news tends to spark counter-replies, creating a dynamic debate. These interaction patterns within the propagation trees offer valuable insights for distinguishing between true and false news. Fake news generally triggers more denying responses, resulting in denser and more complex reply chains. This feedback loop demonstrates how user reactions can amplify or suppress information spread, making it essential to understand these dynamics for effective misinformation detection.

Building on this framework, we propose a novel, weakly supervised method to detect fake news by leveraging Multiple Instance Learning (MIL) [14]. Unlike traditional MIL methods that require labels for individual sentences, our approach uses only document-level veracity annotations, which is more practical for large-scale data. A key challenge is linking these overall news labels to the finer details found in individual sentences and posts within the propagation tree. To address this, we uniquely utilize the propagation trees themselves as the foundational structures for learning stance and veracity patterns from the social media post interactions [15]. To manage the complexity of classifying multiple news and stance types, we decompose the problem into several binary classification tasks. We designate real news and denial stances as "positive" classes at their respective levels, with all other classes treated as "negative." This formulation allows us to train separate MIL models for each veracity-stance pair, effectively capturing the diverse nature of misinformation and user responses. Finally, we introduce a novel hierarchical attention mechanism to integrate the binary model outputs into a coherent multi-class prediction. This mechanism weighs

and combines stance predictions across all MIL models to infer the overall truthfulness of news articles. Additionally, we incorporate a weighted strategy that prioritizes critical sentences during news-level classification, further enhancing prediction accuracy. Altogether, this is the first application of propagation trees within a MIL framework for fake news verification, offering a scalable and insightful approach to understanding and combating misinformation.

The remainder of this paper is organized as follows. Section II reviews related work on fake news detection, including machine learning, deep learning, and social network analysis approaches. Section III defines the problem statement and introduces the formal notations used in our study. Section IV presents the proposed weakly supervised Multiple Instance Learning framework with propagation tree modeling and stance aggregation. Section V describes the construction of the news-post propagation tree and its role in capturing information diffusion. Section VI reports the experimental setup, datasets, evaluation metrics, and results, followed by ablation and component studies. Section VII provides a detailed discussion and interpretation of the findings. Finally, Section VIII concludes the paper and outlines potential directions for future research.

II. Related Work

Fake news detection (FND) research spans three main domains: Machine Learning (ML), Deep Learning (DL), and Social Network Analysis (SNA). ML and DL methods have shown strong performance in identifying fake news through content and user behavior features, while SNA leverages network structures to analyze how misinformation spreads [16]. However, approaches explicitly incorporating social-context models remain limited, highlighting a gap our work addresses. ML techniques typically rely on supervised classifiers like Support Vector Machines (SVM) and Decision Trees (DT). SVMs, using syntactic and lexical features, have achieved F-measures up to 0.87 [17], [18], while DTs and Random Forests (RF) effectively combine content and contextual features, yielding accuracies around 0.86 [19], [20]. Semi-supervised graph-based ML approaches also show promise in detecting fake user accounts spreading misinformation on Twitter, achieving over 90% accuracy [21]. DL methods overcome the manual feature engineering bottleneck of ML by learning hierarchical representations directly from data. Models such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been successfully applied to FND tasks, with hybrid architectures reaching accuracies above 0.8 on benchmark datasets [22], [23], [24]. Geometric deep learning and propagation-based frameworks further capture the relational structure of information diffusion [25]. Comprehensive evaluations demonstrate that ML and DL methods both contribute effectively to fake news detection [26], [27]. SNA approaches analyze the diffusion and interaction patterns within social networks. Large-scale studies leveraging millions of tweets have employed algorithms like Collective Influence to identify super-spreaders of misinformation [28], [29]. Network-based pattern-driven methods [30] and Graph Neural Networks (GNNs) [31] utilize topological and behavioral features to enhance detection accuracy. Knowledge graph techniques enable fact-checking through entity relationships [32], while diffusion network analyses reveal that fake news spreads faster and deeper than truth, amplified by echo chambers and bots [33].

Our approach differs from these prior works by explicitly modeling the social-contextual dynamics of fake news propagation through a novel propagation tree framework that captures user reactions—such as support, doubt, and denial—and their influence on misinformation spread [15]. Unlike conventional ML/DL models focused primarily on content or user-level features, our method leverages weakly supervised Multiple Instance Learning on propagation trees, enabling efficient learning from article-level labels while incorporating the complex interaction patterns inherent in social media discussions. Moreover, we introduce a hierarchical attention mechanism to integrate multi-level stance predictions, improving interpretability and detection accuracy. This social-context-centric approach fills the identified gap in existing research, providing a more holistic understanding and detection of fake news in online networks.

III. Problem Statement

Our strategy for identifying fake news relies on a well-organized dataset denoted as G. Each instance within this dataset, represented by g_i, is a triplet comprising $g_i = \{d_i, C_{d_i}, L_{d_i}\}$ contains the following items:

- News Item (d_i): The central object of investigation, represented as a news article, post, or similar textual content suspected of being fake.
- Replying Posts Set on d_i (C_{d_i}): A chronologically ordered sequence of users' posts on d_i. This sequence captures the surrounding discussion and engagement, providing contextual information for analysis.
- Veracity Label (L_{d_i}): A binary label indicating the truthfulness of the news. $L_{d_i} = 0$ signifies verified truth, while $L_{d_i} = 1$ denotes confirmed falsity.

While C_{d_i} contains all the posts on d_i in temporal order, a crucial aspect lies in acknowledging the explicit connections, such as reply-to and share relationships, that exist between them [34]. We represent each news d_i through a news-post propagation tree (newsPostTree) [15]. The newsPostTree depicts the potential spread of the news from its origin to its wider audience. Edges in this tree follow the direction of potential dissemination. The proposed approach investigates the veracity of fake news on the basis of stances of replying posts on the news. Having stated this, the proposed approach identifies the veracity of fake news on the basis of the following two tasks.

1) Stance detection: Given a post c_i discussing the veracity of a news item d_i, predict the post's overall stance L_{c_i} on the news's accuracy. Possible stances include: *Support:* The post expresses agreement with the news's truthfulness. *Deny:* The post disagrees with the news's accuracy. *Question:* The post raises doubts or seeks clarification about the news's validity. *Comment:* The post focuses on aspects unrelated to the news's veracity.

2) Fake news verification: For each news article d_i, this task aims to determine its overall veracity as either real or fake. We hypothesize that the news article's veracity can be modeled

as a weighted collective assumption based on the stances of its associated posts, similar to the Multiple-Instance Learning (MIL) framework [14]. In our approach, the news article's label is predicted based on the most likely stance category among the stances expressed in the associated posts, considering their distribution within a newsPostTree.

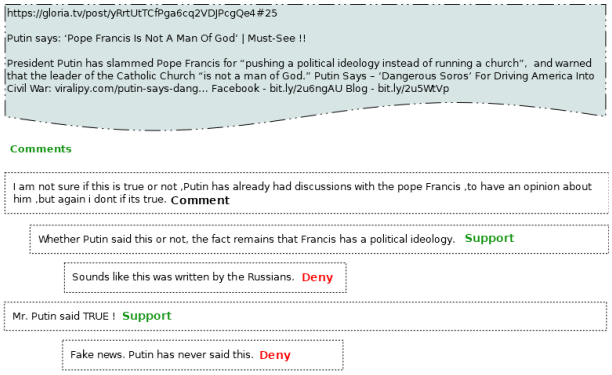


Fig. 1. An example fake news and replying posts on the news.

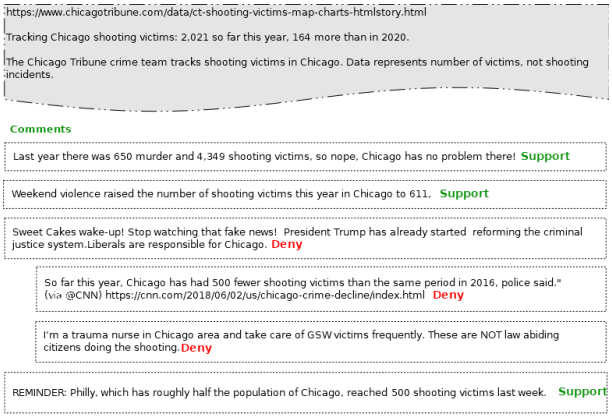


Fig. 2. An example real news and replying post on the news.

IV. Proposed Approach

Traditionally, fake news verification has often been tackled as a single, multi-class classification problem [6]. However, this approach overlooks the intricate interplay between the fine-grained nuances of post stances (e.g. support, deny, question, comment) and the multifaceted nature of news article veracity (e.g. real or fake). To unlock the full potential of this rich information, we propose a novel extension to the Multiple-Instance Learning (MIL) framework. We recognize that a single, multi-class approach might miss the subtle interactions between stance and veracity. Therefore, we decompose the problem into a multitude of binary classification tasks, each focusing on a specific veracity-stance pairing. Let S be the number of stance classes of posts, and let E be the number of veracity classes for news. Given E and S, there exist $H = E * S$ potential veracity-stance target class pairs, each requiring the training of an individual binary classifier for weak supervision in the task of detecting false information. This granularization allows each model to hone its expertise on a precise combination, capturing the subtle details that might be obscured in a global approach.

While individual binary models provide valuable insights, we need to unite their findings to paint a complete picture. This is where the strength of the MIL framework shines. We strategically aggregate the predictions of the individual models, leveraging their collective knowledge to arrive at a robust verdict on the overall news article's veracity.

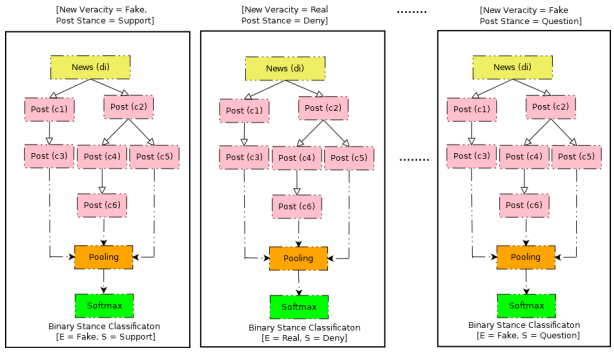


Fig. 3. An architecture of post encoding and binary stance classification of posts.

A. Encoding Replying Posts

To analyze the content of each post and its stance towards news veracity, we leverage powerful neural networks. Each post is first transformed into a sequence of terms $T_{c_i} = \{t_1, t_2, t_3, ..., t_{[c_i]}\}$, a linguistic fingerprint captured by pre-trained term embeddings. These embeddings translate each term t_i into an n-dimensional vector $t_i \in V^n$, encapsulating its meaning and relationships to other terms. Next, we utilize a specialized neural network called a Gated Recurrent Unit (GRU) [35] to understand the flow of information within the post. The GRU processes the term sequence one by one, effectively capturing the context and sentiment woven through the terms. This results in a fixed-size hidden vector that summarizes the essence of the entire post. Finally, to assess the overall stance towards the news article's veracity, we employ two separate GRU-based encoders [36]. These encoders act like dedicated detectives, each scrutinizing the post from a different angle. Their analyses are then condensed into two vectors, capturing the key aspects of the stance of post L_{c_i} and the associated news d_i. In simpler terms, given a number of terms ($T_{c_i} = \{t_1, t_2, t_3, ..., t_{[c_i]}\}$) in the post c_i, the approach breaks down the post into its building blocks (terms) and analyzes how they flow together in a standard GRU transition equation $GRU()$, extracting key aspects of its message (encoders) to ultimately understand its stance towards the news's trustworthiness.

$$V_{d_i} = V_{|d_i|} \tag{1}$$

$$V_{|d_i|} = GRU\left(t_{|d_i|}, V_{|d_i|-1}, \Delta_{d_i}\right) \tag{2}$$

$$V_{c_i} = V_{|c_i|} \tag{3}$$

$$V_{|c_i|} = GRU\left(t_{|c_i|}, V_{|c_i|-1}, \Delta_{c_i}\right) \tag{4}$$

$V_{|d_i|-1}$ and $V_{|c_i|-1}$ serve as hidden units capturing information about the preceding terms in news item d_i and associated post c_i. Δ_{d_i} and Δ_{c_i} encapsulate the entirety of the news and post encoder parameters.

V. News Post Propagation Tree (newsPostTree)

Given a news item d_i, to identify whether d_i is fake news, the proposed approach utilizes the hierarchical tree structure of the replying posts $C_{d_i} = \{c_1, c_2, c_3, ..., c_{|C_{d_i}|}\}$ on d_i, capturing intricate propagation patterns essential for classification (see Fig. 3). Imagine a complex web of online posts, where some spread truth and others spin yarns of misinformation. One such news-post pattern is the "*support-comment-support*" sequence. Imagine some replying posts on a news support the news. Others chime in with questions or comments, and if the evidence in a supporting post holds up, we might see further support emerge. This cyclical pattern tends to be more prevalent in true news, where credible replying posts attract validation and build momentum [15]. On the other hand, fake news often struggle to generate sustained support. They might trigger initial curiosity or even controversy, but without a solid foundation, they tend to fizzle out, lacking the magnetism to attract lasting support.

This model acts like a skilled investigator, navigating this web through the proposed *newsPostTree* structure. Unlike models who focus on individual posts, this model investigates the entire path information takes, starting at the source (root node) and moving down each branch. Each post (node of the tree) can be considered a piece of evidence. This model gathers clues (called "*features*") from each post, like suspicious language or biased references, and strengthens them based on the path they took. If a post agrees with its parent's denial of the original news, the evidence against both strengthens as the model travels down the branch. This "*evidence boost*" happens for each non-leaf node, building a stronger case against potential misinformation. Now, what makes this model special? Unlike models who start at the leaf nodes and piece things together (bottom-up), this model starts at the news (root node) and builds the information tree level by level. Each post's "*evidence file*" combines its own clues with those from its parent, creating a richer picture with each step down the tree. It's like a team of sub-models, each adding their findings to the case news, building a more comprehensive picture with every level [15].

Consider a scenario where the hidden information within a non-leaf node can be seamlessly transmitted to all its child nodes simultaneously, ensuring no loss occurs in the process. In such a context, the determination of the hidden state V_{c_i} for a given node c_i involves a computation that combines the existing hidden state $V_{\text{Parent}(c_i)}$ of its parent node $\text{Parent}(c_i)$ with the input vector specific to the node itself. This integration process allows the formulation of transition equations for node c_i, presenting them in a manner consistent with the principles of a standard Gated Recurrent Unit (GRU).

$$\tilde{V}_{c_i} = V_{c_i}\Gamma \tag{5}$$

$$u_{c_i} = \phi\left(Z_u\tilde{V}_{c_i} + \Upsilon_u V_{Parent(c_i)}\right) \tag{6}$$

$$x_{c_i} = \phi\left(Z_{c_i}\tilde{V}_{c_i} + \Upsilon_x V_{Parent(c_i)}\right) \tag{7}$$

$$\widehat{V}_{c_i} = \tanh\left(Z_V\tilde{V}_{c_i} + \Upsilon_V\left(V_{Parent(c_i)} \odot u_{c_i}\right)\right) \tag{8}$$

$$V_{c_i} = (1 - x_{c_i}) \odot h_{Parent(c_i)} + x_{c_i} \odot \widehat{V}_{c_i} \tag{9}$$

In this context, c_i represents a post, and Γ represents the parameter matrix responsible for transforming this input post. The transformed representation of c_i is denoted as $\tilde{V}c_i$, and Z and Υ signify the weight connections within the GRU (Gated Recurrent Unit). Vc_i refers to the hidden state of c_i, and $\widehat{V}c_i$ represents the candidate activation of the hidden state of the current node. Similar to the standard GRU, $\odot$ indicates element-wise multiplication. The reset gate uc_i determines how to combine the current input $\tilde{V}c_i$ with the memory of children, while the update gate xc_i defines the extent to which memory from the children is incorporated into the current node.

The approach recursively explores the tree, gathering information and "*learning*" about the data point. This information gets condensed into a hidden vector for each leaf. However, the problem is that trees can have different numbers of leaves! This means we can't directly stick those hidden vectors into a neural network with a fixed number of inputs. To solve this, we use a special layer called "*max pooling*" (see Fig. 3). This layer scans all the leaf vectors and picks the highest value for each dimension. This way, we capture the most important features that were highlighted during the walk down the tree, regardless of how many leaves there are. Finally, it considers the best-of-the-best information and feeds it into another layer called "*softmax.*" This layer analyzes the pooled features and makes a prediction about the data point's category. In essence, it takes all the insights gathered from the top-down journey through the tree and uses them to make a confident call. This approach allows us to analyze variable-length trees and effectively leverage the learned features for accurate prediction, even when the journey down the tree can take different paths.

$$L_{d_i} = \text{Softmax}\left(\boldsymbol{W}V_{C_{d_i}} + \boldsymbol{\beta}\right) \tag{10}$$

The parameter $V_{C_{d_i}}$ represents the pooling vector over all leaf nodes, while the parameters W (weight) and β (bias) are instrumental in shaping the output layer.

Our initial investigation explored the feasibility of bottom-up propagation as an alternative information flow mechanism within the model. However, empirical evaluations revealed a substantial decrease in model performance. Analysis identified the overreliance on the root node's representation as the primary culprit for this information loss. This limitation stems from the inherent nature of bottom-up propagation, where information from lower levels aggregates towards the apex, potentially discarding refined features accumulated on diverse propagation paths.

A. Identifying Stance of Posts

Our goal for stance classification is to automatically identify the stance expressed in a social media post, whether it be an opinion or attitude. We can visualize this process as information diffusing through a tree-like structure, with each node representing a component of the post, such as words, phrases, or sentences. Now, think of information flowing down this tree. We assume this diffusion happens perfectly and simultaneously, ensuring all children receive the stance features of their parent. This allows us to build a representation of the overall stance within each "*binary classifier*" $H = E \times S$, which are essentially decision-making units at each leaf node. To achieve this, we used the newsPostTree representation learning approach [37]. This approach works like a translator, converting a node's raw information (including its own content and the stance received from its parent) into a "*context vector.*" This vector captures the gist of the stance expressed at that specific point. Imagine a node c_i, by combining the information from its parent node $Parent(c_i)$ and with its own content, c_i gets translated into a new context vector named $\widehat{V}_{c_i}$. This process repeats recursively as we travel down the tree, with each node's context vector building on the accumulated stance information from its ancestors. In essence, this newsPostTree approach allows us to understand the overall stance of a post by considering not just its individual parts but also the broader context and relationships built up through the hierarchical structure. This information-rich representation empowers the binary classifiers at each leaf node to make more accurate and refined decisions about the specific stance expressed in that part of the text.

$$\widehat{V_{c_i}^h} = \widehat{RNN}\left(V_{c_i}^h, V_{Parent(c_i)}^h, \alpha^h\right) \tag{11}$$

In the context of our model, $\widehat{\text{RNN}}()$ signifies the RNN transition function based on top-down processing [37] of the newsPostTree, with α^h encompassing all associated parameters. Subsequently, a fully-connected softmax layer is employed to forecast the stance probability of c_i concerning the news vector $V_{d_i}^h$ in relation to classifier h:

$$p_{c_i}^h = \text{softmax}\left(\delta_1^h \widehat{V_{c_i}^h} + \delta_2^h V_{d_i}^h + \omega^h\right) \tag{12}$$

In this formulation, δ_1^h, δ_2^h, and ω^h denote the weights and bias of the prediction layer. Subsequent to this, the stance probability for each individual post $p_{c_i}^h$ within the newsPostTree can be calculated using a methodology akin to that described in the above equation.

B. Post-Stance Aggregation

In our quest to assess fake news veracity, not all pieces of evidence hold equal weight. Consequently, it's crucial to selectively amplify the contributions of those pieces of evidence – the binary classifiers within the tree – that demonstrate the strongest correlation between the expressed stances and the actual news veracity. To achieve this, we enlist the power of an "*attention mechanism*" [38]. It carefully scans all the binary classifiers within the tree, analyzing their individual capabilities in capturing the link between stance and news veracity. Those classifiers exhibiting a stronger correlation indicate greater reliability as evidence sources. Conversely, classifiers with weaker correlations fade into the background, their contributions minimized.

$$V_a = GRU\left(t_{|d_i|}, V_{|d_i|-1}, \Delta_a\right) \tag{13}$$

$$\Lambda_h = \frac{\exp\left(V_a \cdot V_{d_i}^{h^\top}\right)}{\sum_h \exp\left(V_a \cdot V_{d_i}^{h^\top}\right)} \tag{14}$$

Δ_a symbolizes the internal configuration of the GRU encoder. $V_{c_i}^h$ represents the news encoded by the h-th classifier. To understand the different stances, our model relies on a collection of specialized "*binary stance classifiers,*" each focusing on a particular stance type. We aggregate all binary stance classifiers targeting the same stance type into a single set $K(S)$, where $s \in S$ can be support, deny, comment, or question. So, while they're all investigating the same type of news, they each have a unique perspective based on their specific training data. The final probability for that stance type is then computed using:

$$\widehat{p_{c_i,S}} = \sum_{h \in K(S)} \Lambda_h \cdot p_{c_i}^h \tag{15}$$

To assess the overall stance landscape, let $K(S)$ represent a team of classifiers focusing on a specific stance type $S \in \{deny, support, comment, question\}$. Each classifier within $K(S)$ outputs a prediction ($p_{c_i}^h$) the probability that the post c_i expresses the target stance S through the stance classifier h. Consequently, $p_{c_i,S}$ denotes the probability that post c_i is categorized as stance S, taking into account the significance of the corresponding classifiers with S as the target class. Despite addressing the same stance, these classifiers vary because each concentrates on a distinct veracity label of the news. By accumulating the weighted predictions for the specific stance S, we obtain the final probability distribution $\widehat{p_{c_i}}$. This process repeats for all possible stance types in S, ultimately painting a comprehensive picture of the different opinions and their relative strengths within the post.

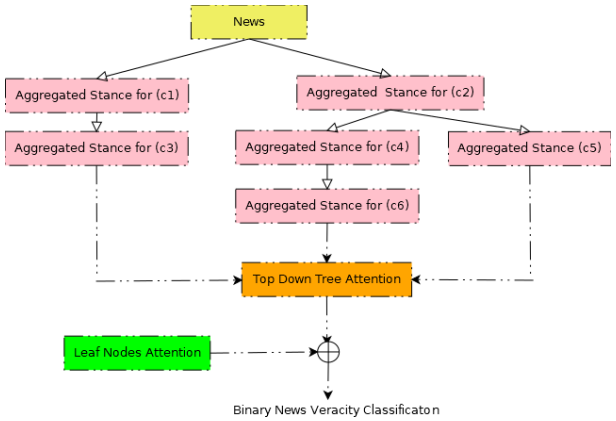


Fig. 4. An architecture of identifying fake news.

C. Identifying Fake News

In the context of verifying news authenticity, simply taking the overall sentiment of a discussion about the news might not be enough. We need to delve deeper, analyzing the individual stances expressed at different replying posts in the conversation and prioritizing those carrying greater evidential weight. To achieve this, we propose a "*newsPostTree Attention Mechanism*" that selectively aggregates stance information throughout the discussion tree (see Fig. 4). Think of the discussion as a branching tree, where each node represents a stance towards the news. Our model focuses on attending to specific nodes within each branch – the ones offering strong evidence for or against the news's veracity. We call these "*evidential stance nodes.*" Formally, let's call a path β_l through the tree from its root d_i (news) to a leaf node c_i. The set of all nodes along that path is denoted by C_{β_l}. To determine the overall aggregated stance for the path β_l, our model accomplishes this in two steps:

1) Selective attention: For each path, the model focuses on the evidential stance nodes. It assigns attention scores to each node, essentially highlighting the ones carrying the most significant evidence for or against the news's veracity. This selective attention mechanism ensures that the aggregated stance doesn't get diluted by less informative opinions.

2) Aggregation: Once the attention scores are calculated, the model combines the information from the evidential stance nodes for the path. This aggregation, using the attention scores as weights, produces a single "*aggregated stance*" for the path. This stance reflects the dominant sentiment, filtered through the lens of the most valuable pieces of evidence identified along the path.

$$\Theta^h_{c_i} = \frac{\exp\left(\widehat{V^h_{c_i}} \cdot {V^h_{d_i}}^\top\right)}{\sum_{i \in C_{\beta_l}} \exp\left(\tilde{V}^h_{d_i} \cdot {V^h_{d_i}}^\top\right)} \tag{16}$$

$$p^h_{\beta_l} = \sum_{n \in C_{\beta_l}} \Theta^h_{c_n} \cdot p^h_{c_n} \tag{17}$$

The symbol $\Theta^h_{c_i}$ denotes the attention coefficient assigned to each node along the path β_l, while $p^h_{\beta_l}$ represents the aggregated probability of stances from leaf nodes along the propagation path β_l. The symbiotic relationship between $\Theta^h_{c_i}$ and $p^h_{\beta_l}$ is pivotal, as it quantifies attention given to individual nodes and determines the collective stance probability of leaf nodes along the specific propagation path.

To capture the overall sentiment surrounding the news, we need to further aggregate information across different paths. Here's where the second stage of our Tree Attention Mechanism starts its work, focusing on informative path selection. Imagine each possible discussion about the news (a path through the tree) as a potential piece of evidence. We want to pick the paths that offer the most valuable insights into the news's veracity. To do this, we leverage the hidden vectors V^k_l embedded in the leaf nodes of each path (β_l). These vectors, generated in the first stage, encapsulate the accumulated stance information at the end of each branch.

Formally, let $\widehat{C_{d_i}}$ represent the set of all leaf nodes within the entire tree, with its root being the news itself. We now employ the Tree Attention Mechanism again, but this time on the set $\widehat{C_{d_i}}$. This second-stage attention assigns scores to each individual path based on the informativeness of its leaf node's hidden vector. Paths with stronger evidence, reflected in their leaf node representation, receive higher scores and are ultimately weighted more heavily in the final aggregation. This weighted aggregation, considering the attention scores for each path, culminates in an "*aggregated path stance.*" This stance represents the collective opinion about the news, filtered through the lens of the most informative paths identified by the second-stage attention mechanism. By prioritizing paths conveying robust evidence, we ensure that our final verdict on the news's veracity is not swayed by less insightful discussions or irrelevant tangents.

$$\Theta^h_{\beta_l} = \frac{\exp\left(\widehat{V^h_{\beta_l}} \cdot {h^h_{d_i}}^\top\right)}{\sum_{\beta_l \in \widehat{C_{d_i}}} \exp\left(\widehat{V^h_{\beta_l}} \cdot {h^h_{d_i}}^\top\right)} \tag{18}$$

$$\widehat{p^h_{d_i}} = \sum_{\beta_l \in \widehat{C_{d_i}}} \Theta^h_{\beta_l} \cdot p^h_{\beta_l} \tag{19}$$

The $\Theta^h_{\beta_l}$ signifies the attention coefficient for each individual leaf node, while $\widehat{p^h_{d_i}}$ encapsulates the probability associated with fake news veracity, derived from aggregating stances along various paths.

D. Aggregating Fake News Classifiers

Instead of analyzing each fake-news veracity classifier individually, our model employs a clever strategy to leverage their collective wisdom. Similar to stance classifiers aggregation, we gather all classifiers targeting the same "*fake news class label*" into a single set. Within each set, the members, which are individual binary veracity classifiers, provide their own assessments of the news's truthfulness. To combine these diverse insights, we utilize a "*weighted sum*" approach.

We used an indicator set $K(e_o)$ to group binary classifiers targeting the same veracity class e_{fake}, e_{real} in E. Each classifier within $K(e_o)$ provides a binary veracity prediction (fake or real) for the news item d_i, representing the probability $p^h_{d_i}$ of it belonging to veracity class e_o through the classifier h. Notably, although they share the same veracity target, classifiers in $K(e_o)$ differ in their assigned "*target stance classes.*" These stance classes represent specific sentiment or opinion types they analyze within the news. This differentiation leads to diverse stance probability predictions across classifiers, even for the same veracity class. By aggregating these stance-informed predictions, we obtain a richer representation of the news item's veracity. The final probability distribution over all veracity classes, denoted by $\widehat{p_{d_i}} = [\widehat{p_{d_i,fake}}, \widehat{p_{d_i,real}}]$, captures the combined insights from each classifier's unique perspective.

E. Training and Optimizing the Learning Model

To train our model using multiple binary classifiers, we convert the fine-grained veracity and stance annotations into binary labels. Each classifier targets a specific veracity-stance pair, where the veracity label of a news article d_i is mapped to either the target class or *others* (non-matching cases). Similarly, each post's stance is modeled as a probability distribution over the target stance class. This label transformation is applied uniformly across all classifiers, enabling them to distinguish between a specific veracity-stance combination and all other instances. While stance predictions are utilized during training, the ground truth for fake news verification focuses solely on the veracity of the news article, as post-level stance annotations are often sparse and potentially unreliable.

1) Training binary MIL-based classifiers: Our model hinges on the negative log likelihood (NLL) as the key metric for measuring prediction accuracy.

$$M = -\sum_{h=1}^{H}\sum_{d_i=1}^{D} L_{d_i} * \log \hat{p}_{d_i}^{h} + (1 - L_{d_i}) * \log\left(1 - \hat{p}_{d_i}^{h}\right) \quad (20)$$

For each news item d_i, let L_{d_i} represent the ground truth. $p_{d_i}^{h}$, on the other hand, signifies the predicted probability generated by classifier h for the news item d_i. This equation takes into account all D news items and H binary classifiers in the model, making it a powerful tool for evaluating how well predicted probabilities align with actual outcomes across the entire dataset.

2) Training aggregation model: To guide our aggregation model towards optimal performance, we leverage a powerful training technique called the "*negative log-likelihood loss function.*"

$$\widehat{M} = -\sum_{f=1}^{F}\sum_{e=1}^{E} L_{f,e} * \log \hat{p}_{f,e} + (1 - L_{f,e}) * \log\left(1 - \hat{p}_{f,e}\right) \quad (21)$$

The model produces a "*predicted probability*" ($p_{f,e}$) of the f news belonging to a specific veracity class $e \in E$. In this context, $L_{f,e}$ represents a binary value that signifies whether the veracity class of the f-th news, based on groundtruth, is e. Finally, E signifies the total number of possible veracity classes.

The negative log-likelihood loss function then steps in with its rigorous assessment. It calculates a penalty based on the discrepancy between the model's prediction and the actual truth. If the model confidently predicts the wrong class, the penalty is severe, urging it to adjust its reasoning. Conversely, a confident and accurate prediction earns minimal punishment. As the model processes numerous news items, it undergoes a continuous learning process guided by the penalties. It refines its ability to interpret and weigh the individual stance information, gradually honing its skill in predicting the overall veracity of the news with increasing accuracy. By minimizing the negative log-likelihood over countless news samples, the model effectively learns to distinguish fake news news from real news within the chosen framework of veracity classes.

To fine-tuning We used "*back-propagation*" [39], which meticulously analyzes the model's output and works backwards to optimize the model. To guide this optimization process, we leverage the "*Adam*" optimizer [40]. which maneuvering the learning rate (the pace of adjustments) to ensure the model progresses efficiently without getting stuck in overfitting or underfitting traps. We set the initial learning rate at 0.001, and it can dynamically adapt based on the model's performance. To infuse our model with rich linguistic understanding, we pre-train its word embedding layer using pre-trained "*GloVe Wikipedia 6B word embeddings*" [41]. These pre-trained vectors offer an initial advantage, enabling the model to swiftly understand the intricacies of language within the news it analyzes. The training process persists until two crucial conditions are satisfied:

- Convergence of the loss value: We monitor the "*loss function,*" a metric that quantifies the model's errors. When the loss stabilizes and stops decreasing significantly, it indicates that the model has likely reached its optimal performance.
- Maximum epoch number reached: In case the loss doesn't converge perfectly, we also set a pre-defined maximum number of training iterations ("*epochs*") as a safety net. This ensures the training process doesn't run indefinitely and allows us to evaluate the model's performance at various stages.

At the end the model applies the "*weighted aggregation model,*" which combines the insights from all individual classifiers to deliver a unified verdict on the news item's overall veracity.

VI. Experiments

For investigating the impact of news propagation patterns on fake news detection, we used the FakeNewsNet datasets [42], which are integral to understanding the dynamics of misinformation in online spaces. These datasets encompass source news that has undergone fact-checking by reputable sources such as PolitiFact and GossipCop, along with the associated tweets and social engagement data of users. The PolitiFact dataset comprises a total of 1,056 news articles, including 432 identified as fake news and 624 classified as real news articles. Similarly, the GossipCop dataset encompasses a total of 22,140 news articles, with 5,323 categorized as fake news and 16,817 as real news articles. Regarding the social context, PolitiFact observed 95,553 users posting tweets about fake news and 249,887 users posting tweets about real news. In the case of GossipCop, fake news generated 265,155 user tweets, while real news garnered 80,137 user tweets. The user social data is comprehensive, involving the crawling of user attributes and user-following relations, providing insights into how individuals interact with and disseminate information. The labels within the dataset categorize news as either "*real news*" or "*fake news*". To ensure a focused analysis, retweets are filtered out from the datasets, allowing a more clear examination of the primary propagation patterns.

We trained and tested our model on the SemEval-8 dataset [43]. SemEval-8 offers a rich tapestry of data, with each news item meticulously labeled as either "*true,*" "*fake,*" or "*unverified-fake.*" We consider the samples only labeled

as either "*true*" or "*fake*". This veracity labeling goes beyond a simple binary classification, offering valuable insights into the spectrum of information trustworthiness. Moreover, each news item is accompanied by user posts categorized by their expressed stance: "*support,*" "*deny,*" "*question,*" or "*comment.*" While prior research on SemEval-8 often focused on a holistic understanding of rumor detection and stance prediction, our investigation took a more granular approach. We focused exclusively on news items categorized as true or fake, effectively framing the task as a binary veracity classification. This decision allowed us to hone in on the core challenge of distinguishing factually accurate news from misinformation. Furthermore, instead of simply predicting the overall stance category for each post, we opted for a more fine-grained analysis. We trained eight individual binary stance classifiers, one for each possible stance-veracity combination.

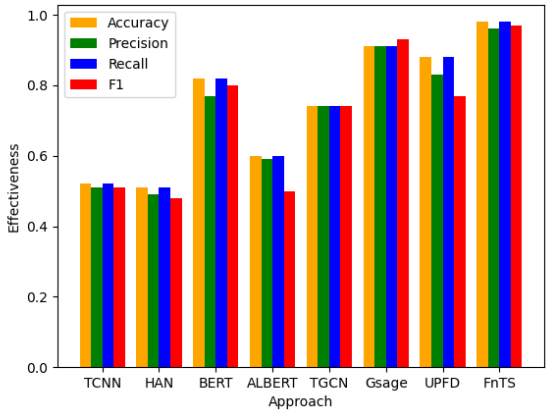
To optimize our model's performance, we held out 20 percent of the test data as a validation set. This allowed us to adjust hyperparameters precisely, ensuring the best possible performance. We further employed a rigorous set of evaluation metrics, including Accuracy, Precision, Recall, and F1 score (F1). These metrics, taken together, offer a comprehensive picture of how effectively our models distinguish real and fake news, while also capturing the fine details of stance detection. Furthermore, we built all our neural networks using the well-regarded PyTorch framework [44]. This ensured consistency and reliability throughout the computational aspects of the study.

A. Results

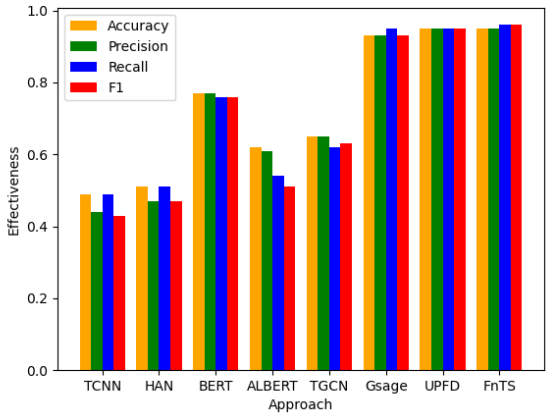
To evaluate the efficacy of any novel model demands a rigorous testing ground. For the proposed approach, we constructed a two-pronged evaluation strategy. First, we compare the performance of our proposed approach (**FnTS**) against several baseline approaches: TextCNN [45], HAN [46], BERT [47], ALBERT [48], TextGCN [49], GraphSage [50], and UPFD [51]. Second, we armed ourselves with a quartet of robust evaluation metrics: Accuracy, Precision, Recall, and F1-score (F1). These metrics collectively paint a clear picture of how effectively FnTS discerns real from fake news, capturing both its overall success rate and its ability to avoid both false positives and negatives.

According to Fig. 5, results on both datasets show that the proposed approach (FnTS) achieves the optimal score on all four metrics. This consistent performance across metrics and datasets underscores the strength of our approach, suggesting that considering the stance of posts truly makes a difference in the war against misinformation. Further investigation unveils the secret sauce behind FnTS's success: its top-down exploration of social networking posts. By delving deeper into the intricacies of social connections, the newsPostTree hierarchy exploration captures a richer tapestry of information compared to baseline methods. Specifically, the attention mechanism, a built-in spotlight highlighting vital post-stances, plays a crucial role in identifying and extracting distinctive user signals. These signals ultimately empower FnTS to make more informed and accurate fake news detection decisions.

Graph models, which utilize both textual and structural information, consistently surpass their counterparts that focus solely on text. Notably, GraphSage [50], powered by a customized news propagation graph, consistently demonstrates outstanding performance, outperforming all text-based methods. On the other hand, UPFD [51], another strong competitor, also uses user information as a tool. However, its reliance on historical data makes its performance vulnerable to fluctuations specific to the dataset, unlike the steadfast accuracy displayed by FnTS. To conclude, the comprehensive evaluation of FnTs paints a compelling picture of its effectiveness. Not only does it outperform or closely match baseline approaches, but it also sheds light on the power of incorporating user connections into fake news detection models. These findings pave the way for further advancements in leveraging social network dynamics to combat the ever-evolving landscape of misinformation.



(a) Politifact.

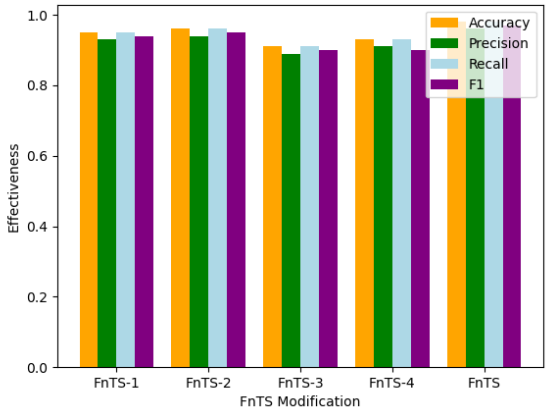


(b) Gossipcop.

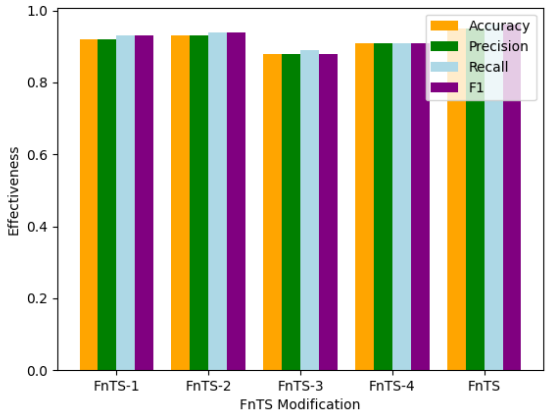
Fig. 5. Effectiveness of the proposed approach (FnTS) and baseline approaches on two benchmark datasets.

B. Component Study

To understand how each components of our proposed approach contributes to its success in fighting fake news, we conducted an experiment called an "*ablation study*" on FnTS with four modifications.



(a) Politifact.



(b) Gossipcop.

Fig. 6. Component studies of the proposed approach.

- FnTS-1: In this modification, we kept the tree-based attention mechanism but replaced the "*post encoder*" with non-structured post encoder.
- FnTS-2: We replaced the "*newsPostTree attention mechanism*" with a more common one (dot product) for stance aggregation.
- FnTS-3: In this modification we remove the recursive tree-based stance aggregation.
- FnTS-4: In this modification we remove the tree-based attention mechanism and replace the tree-based post encoder with a non-structural post encoder.

The results, shown in Fig. 6, indicate that FnTS-3 performs the worst, showing that the tree structure in both post encoding and stance aggregation is crucial. Without it, the model's ability to spot fake news drops significantly. FnTS-4 also stands out with the lowest performance scores, displaying substantial declines in accuracy, precision, recall, and F1 score for fake news verification. Changes to the tree-based attention mechanism (FnTS-2/FnTS-4) also hurt performance the most. This tells us that how the model focuses on its relevant parts while understanding their relationships through their tree structure is critical for its accuracy. In simpler terms, our experiment reveals that both the tree structure and the special way the model pays attention to different parts of the posts are essential for its success in detecting fake news. By understanding these key components, we can further improve our model and keep up with the ever-evolving world of misinformation.

TABLE I. ATTENTION SCORES OF BRANCHES OF NEWS-POST PROPAGATION TREE FOR CLASSIFYING FAKE NEWS

Branch	Attention Score
News− >Support− >Query− >Support	0.12
News− >Support− >Query− >Comment	0.07
News− >Support− >Support	**0.46**
News− >Deny− >Deny	**0.20**
News− >Comment− >Support− >Support	**0.21**
News− >Comment− >Comment	0.02
News− >Comment− >Support	0.04
News− >Query− >Comment	0.03
News− >Query− >Deny	0.02

TABLE II. ATTENTION SCORES OF BRANCHES OF NEWS-POST PROPAGATION TREE FOR CLASSIFYING REAL NEWS

Branch	Attention Score
News− >Comment− >Deny− >Support	0.12
News− >Comment	0.02
News− >Comment− >Deny	0.09
News− >Comment− >Deny− >Support	**0.15**
News− >Query− >Deny	0.09
News− >Deny− >Support	**0.43**
News− >Deny− >Support− >Comment	**0.23**

VII. DISCUSSION ON THE PROPOSED APPROACH

To peek into the inner workings of the proposed approach, we designed an experiment using the FakeNewsNet datasets [42]. We selected two trees from this dataset, one where the source news is "*Real*" and another where it's "*Fake.*" Table I and Table II show how our model analyzes the comments and replies associated with each news, represented as branches and leaves in the "*trees.*" The bold paths highlight the tree branches that are deemed most relevant for determining whether the news is actually true or false. By looking at the attention scores assigned to each post, we can uncover fascinating patterns:

1) Fake news: When the source news is labeled "*Fake,*" the model focuses heavily on comments that deny the news and then support it (branches with "*deny-support*" and "*deny-support-comment*"). This "*doubling back*" behavior suggests the model identifies these conflicting stances as suspicious clues.

2) Real news: On the other hand, for "*Real*" news, the model prioritizes comments that consistently support the news or deny the deny post (branches with "*support-support,*" "*deny-deny,*" and "*comment-support-support*"). This aligns with our intuition that consistent stances reinforce the legitimacy of the news.

3) Deny vs. Support: Interestingly, comments expressing denial play a more significant role when the news is classified as "*Fake*" (higher attention scores on denial branches). Conversely, support comments hold greater weight when the news is deemed "*Real.*"

4) Propagation paths: The branches trace the pathways through the trees that the model deems most influential in reaching its verdict. This highlights the importance of considering the structure of the discussion along with the individual comments themselves.

These observations confirm our belief that the "*tree propagation structure*" is vital for accurately recognizing fake news. By analyzing the relationships between comments and their stances, the proposed approach builds a nuanced understanding of online discourse, ultimately separating fact from fiction in the ever-evolving world of information.

VIII. Conclusion

This research presents a novel weakly supervised propagation model built upon the Multiple Instance Learning (MIL) framework. The proposed model is designed to tackle the dual challenge of simultaneously verifying the authenticity of fake news and identifying the stances expressed in associated posts. Crucially, our model achieves this feat using only bag-level annotations, specifically fake news veracity labels, demonstrating its ability to effectively learn from limited supervision. This unique approach empowers the model to collectively infer both the truthfulness of news and the previously unseen stance labels for individual posts within the associated propagation tree. The distinct tree-based stance aggregation mechanisms, deployed using newsPostTree configurations, are a key innovation of our work. This mechanism has yielded demonstrably superior performance for fake news identification tasks, surpassing the benchmarks established by state-of-the-art supervised and unsupervised models. The significance of our findings resides in the successful application of weakly supervised learning to significantly enhance the accuracy and efficiency of fake news verification in social media settings. The proposed framework opens several avenues for future work. One promising direction is extending the model to handle multi-modal data by incorporating images, videos, and metadata alongside textual and propagation features. Another important line of research is applying the framework to emerging real-time scenarios, such as early detection of misinformation during breaking news events. Additionally, improving interpretability through explainable AI techniques would make the system more transparent and valuable for journalists, fact-checkers, and policymakers.

Acknowledgment

This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU).

References

[1] Z. Jin, J. Cao, Y.-G. Jiang, and Y. Zhang, "News credibility evaluation on microblog with a hierarchical propagation model," in *2014 IEEE International Conference on Data Mining*. IEEE, 2014, pp. 230–239.

[2] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–36, May 2017. [Online]. Available: https://www.aeaweb.org/articles?id=10.1257/jep.31.2.211

[3] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu, "defend: Explainable fake news detection," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 395–405.

[4] J. Alghamdi, S. Luo, and Y. Lin, "A comprehensive survey on machine learning approaches for fake news detection," *Multimedia Tools and Applications*, vol. 83, no. 17, pp. 51 009–51 067, 2024.

[5] V.-H. Nguyen, K. Sugiyama, P. Nakov, and M.-Y. Kan, "Fang: Leveraging social context for fake news detection using graph representation," in *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020, pp. 1165–1174.

[6] K. Pelrine, J. Danovitch, and R. Rabbany, "The surprising performance of simple baselines for misinformation detection," in *Proceedings of the Web Conference 2021*, 2021, pp. 3432–3441.

[7] R. K. Kaliyar, A. Goswami, P. Narang, and S. Sinha, "Fndnet–a deep convolutional neural network for fake news detection," *Cognitive Systems Research*, vol. 61, pp. 32–44, 2020.

[8] Y. Liu and Y.-F. Wu, "Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[9] X. Ge, S. Hao, Y. Li, B. Wei, and M. Zhang, "Hierarchical co-attention selection network for interpretable fake news detection," *Big Data and Cognitive Computing*, vol. 6, no. 3, p. 93, 2022.

[10] N. Rosenfeld, A. Szanto, and D. C. Parkes, "A kernel of truth: Determining rumor veracity on twitter by diffusion pattern alone," in *Proceedings of The Web Conference 2020*, 2020, pp. 1018–1028.

[11] B. Xie, X. Ma, J. Wu, J. Yang, and H. Fan, "Knowledge graph enhanced heterogeneous graph neural network for fake news detection," *IEEE Transactions on Consumer Electronics*, 2023.

[12] E. Min, Y. Rong, Y. Bian, T. Xu, P. Zhao, J. Huang, and S. Ananiadou, "Divide-and-conquer: Post-user interaction network for fake news detection on social media," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 1148–1158.

[13] A. Jarrahi and L. Safari, "Evaluating the effectiveness of publishers' features in fake news detection on social media," *Multimedia Tools and Applications*, vol. 82, no. 2, pp. 2913–2939, 2023.

[14] J. Foulds and E. Frank, "A review of multi-instance learning assumptions," *The knowledge engineering review*, vol. 25, no. 1, pp. 1–25, 2010.

[15] R. Yang, J. Ma, H. Lin, and W. Gao, "A weakly supervised propagation model for rumor verification and stance detection with multiple instance learning," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1761–1772.

[16] F. Olan, U. Jayawickrama, E. O. Arakpogun, J. Suklan, and S. Liu, "Fake news on social media: the impact on society," *Information Systems Frontiers*, vol. 26, no. 2, pp. 443–458, 2024.

[17] H. Zhang, Z. Fan, J. Zheng, and Q. Liu, "An improving deception detection method in computer-mediated communication," *Journal of Networks*, vol. 7, no. 11, p. 1811, 2012.

[18] F. Fkih, D. Rhouma, and H. Alghofaily, "A semantic approach for sarcasm identification for preventing fake news spreading on social networks," *International Journal of Information Technology*, pp. 1–19, 2024.

[19] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th international conference on World wide web*, 2011, pp. 675–684.

[20] E. J. Briscoe, D. S. Appling, and H. Hayes, "Cues to deception in social media communications," in *2014 47th Hawaii international conference on system sciences*. IEEE, 2014, pp. 1435–1443.

[21] M. Balaanand, N. Karthikeyan, S. Karthik, R. Varatharajan, G. Manogaran, and C. Sivaparthipan, "An enhanced graph-based semi-supervised learning algorithm to detect fake users on twitter," *The Journal of Supercomputing*, vol. 75, pp. 6085–6105, 2019.

[22] O. Ajao, D. Bhowmik, and S. Zargari, "Fake news identification on twitter with hybrid cnn and rnn models," in *Proceedings of the 9th international conference on social media and society*, 2018, pp. 226–230.

[23] W. Y. Wang, "" liar, liar pants on fire": A new benchmark dataset for fake news detection," *arXiv preprint arXiv:1705.00648*, 2017.

[24] A. Zubiaga, M. Liakata, R. Procter, G. Wong Sak Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PloS one*, vol. 11, no. 3, p. e0150989, 2016.

[25] S. M. TS and P. Sreeja, "Fake news detection on social media using adaptive optimization based deep learning approach," *Multimedia Tools and Applications*, pp. 1–21, 2024.

[26] S. S. V. U, P. S. R, S. M, and S. R.B.V., "I-s2fnd: a novel interpretable self-ensembled semi-supervised model based on transformers for fake news detection," *J. Intell. Inf. Syst.*, vol. 62, no. 2, p. 355–375, Oct. 2023. [Online]. Available: https://doi.org/10.1007/s10844-023-00821-0

[27] J. Y. Khan, M. T. I. Khondaker, S. Afroz, G. Uddin, and A. Iqbal, "A benchmark study of machine learning models for online fake news detection," *Machine Learning with Applications*, vol. 4, p. 100032, 2021.

[28] K. Shu, H. R. Bernard, and H. Liu, "Studying fake news via network analysis: detection and mitigation," *Emerging research challenges and opportunities in computational social network analysis and mining*, pp. 43–65, 2019.

[29] A. Bovet and H. A. Makse, "Influence of fake news in twitter during the 2016 us presidential election," *Nature communications*, vol. 10, no. 1, p. 7, 2019.

[30] X. Zhou and R. Zafarani, "Network-based fake news detection: A pattern-driven approach," *ACM SIGKDD explorations newsletter*, vol. 21, no. 2, pp. 48–60, 2019.

[31] Y. Han, S. Karunasekera, and C. Leckie, "Graph neural networks with continual learning for fake news detection from social media," *arXiv preprint arXiv:2007.03316*, 2020.

[32] G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini, "Computational fact checking from knowledge networks," *PloS one*, vol. 10, no. 6, p. e0128193, 2015.

[33] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, "Fake news on twitter during the 2016 us presidential election," *Science*, vol. 363, no. 6425, pp. 374–378, 2019.

[34] X. Su, J. Yang, J. Wu, and Y. Zhang, "Mining user-aware multi-relations for fake news detection in large scale online social networks," in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 2023, pp. 51–59.

[35] R. Dey and F. M. Salem, "Gate-variants of gated recurrent unit (gru) neural networks," in *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*. IEEE, 2017, pp. 1597–1600.

[36] J. Zhang, J. Du, and L. Dai, "A gru-based encoder-decoder approach with attention for online handwritten mathematical expression recognition," in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 902–907.

[37] J. Ma, W. Gao, and K.-F. Wong, "Rumor detection on twitter with tree-structured recursive neural networks." Association for Computational Linguistics, 2018.

[38] A. de Santana Correia and E. L. Colombini, "Attention, please! a survey of neural attention models in deep learning," *Artificial Intelligence Review*, vol. 55, no. 8, pp. 6037–6124, 2022.

[39] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of machine learning research*, vol. 12, no. ARTICLE, pp. 2493–2537, 2011.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[41] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[42] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media," *Big data*, vol. 8, no. 3, pp. 171–188, 2020.

[43] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, "Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017.

[44] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[45] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[46] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[47] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.

[48] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in *International Conference on Learning Representations*, 2019.

[49] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 7370–7377.

[50] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[51] X. Su, J. Yang, J. Wu, and Y. Zhang, "Mining user-aware multi-relations for fake news detection in large scale online social networks," in *Proceedings of the sixteenth ACM international conference on web search and data mining*, 2023, pp. 51–59.

Hierarchical Adaptive Gap-Run TID Compression for Large-Scale Frequent Itemset Mining

XIN DAI[1], CHENJIAO LIU[2], XUE HAO[3], QICHEN SU[4]
Faculty of Computing, Universiti Teknologi Malaysia (UTM), Jalan Iman, 81310 Skudai, Johor Bahru, Malaysia[1,3,4]
School of Mathematics and Big Data, Guizhou Education University, 115 Gaoxin Road, Guiyang, Guizhou 550018, China[1]
Guizhou Key Laboratory of Artificial Intelligence and Brain-inspired Computing,
Guizhou Education University, 115 Gaoxin Road, Guiyang, Guizhou 550018, China[2]

***Abstract*—Frequent itemset mining faces the prominent problems of high storage space requirements and low efficiency in a large-scale transaction data environment.The traditional Eclat algorithm usually uses bitmap or sparse array to represent a single transaction identifier (TID), which is difficult to adapt to the changes of dense and sparse transaction data at the same time; Although the existing hybrid representation schemes can partly alleviate this problem, the additional computational overhead caused by frequent data structure switching and the inherent space waste of bitmap structure have not been fundamentally solved. Therefore, this article proposes a HiAGL-FIM algorithm based on Hierarchical Adaptive Gap Run Transaction Identifier List (HAGL-TID). This algorithm adaptively selects Gap List or Run List for transaction identifier encoding through continuity ratio, and designs an efficient TID intersection operation method, completely eliminating dependence on bitmap structure and effectively reducing memory consumption and intersection calculation overhead. The experimental results show that HiAGL-FIM has significant advantages in terms of running time, memory usage, and data scalability compared to classical algorithms such as Eclat, FP Growth, and dEclat. Especially when the transaction data scale reaches millions, it shows a more significant performance improvement, demonstrating the effectiveness and practical value of our method.**

***Keywords*—Frequent itemset mining; pure Eclat; Hierarchical Adaptive Gap-Run List (HAGL-TID); large-scale transaction data**

I. Introduction

With the rapid development of big data technology and artificial intelligence, frequent itemset mining (FIM), as one of the core problems in the field of data mining, has received widespread attention and in-depth research in recent years [1]. The goal of frequent itemset mining is to identify frequently occurring combinations of items from transactional datasets, which plays an important role in many application fields such as market basket analysis, recommendation systems, bioinformatics, and anomaly detection[2]. Among the existing frequent itemset mining methods, Apriori algorithm and FP Growth algorithm are widely used due to their clear theoretical basis and simple implementation [3]. However, the Apriori algorithm has low efficiency due to its multiple scans of the database and frequent generation of candidate sets; Although FP Growth avoids frequent database scans, it may incur significant memory overhead when building FP trees [4]. The Eclat algorithm records itemset support information through a Transaction Identifier List (TID), significantly reducing the number of database scans and achieving good efficiency [5]. However, traditional Eclat algorithms usually only use a single TID representation method (such as sparse arrays or bitmaps), which fails to simultaneously consider the sparse and continuous features of transaction data, resulting in low memory efficiency and significant optimization space for performance when applied in real datasets [6].

In recent years, researchers have proposed various hybrid TID representation strategies around "density adaptation". The typical approach is to dynamically switch between bitmap, array, or N-list/NegNodeset structures based on local transaction density, such as Tseng et al.'s FPL/TPL adaptive framework [7], Poovan et al.'s multi-threaded NegNodeset+N-list hybrid framework [8], and Bashir and Baig's HybridMiner algorithm [9]. These methods have achieved a good trade-off between time and space on small and medium-sized datasets, but there are still two major bottlenecks: firstly, the threshold for structure switching depends on empirical settings, which requires repeated re-encoding of TID in the face of frequent fluctuations in transaction density in streaming scenarios, resulting in significant CPU and cache overhead; secondly, even though bitmap is efficient in dense scenes, its storage requirements still increase linearly with transaction volume, resulting in a large amount of empty space waste when sparsity increases. The Ramp algorithm has conducted a detailed analysis on this [10]. Furthermore, the closed set miner CICLAD, which targets data streams of millions or more, also reported the issue of high peak memory consumption in traditional bitmap/node-set methods [11]. Therefore, how to completely eliminate the reliance on bitmap representations while enabling finer-grained continuity modeling, and how to maintain stable mining efficiency in environments with sparse remains a major gap in current mixed TID research. This issue is mainly reflected in two aspects: first, existing hybrid approaches depend heavily on frequent structure switching and empirically set thresholds, which often require repeated re-encoding under fluctuating transaction densities and lead to additional CPU and cache overhead; second, the storage cost of bitmap structures increases linearly with data size, causing severe space waste in sparse regions and making it difficult to meet the performance demands of large-scale transactional datasets.

In response to the above challenges, this article proposes a new frequent itemset mining algorithm — HiAGL-FIM. This algorithm innovatively introduces a transaction-ID representation method based on Hierarchical Adaptive Gap Run List (HAGL-TID). The HiAGL-FIM algorithm does not rely on traditional bitmap structures; instead, it adaptively selects Gap or Run lists for compression storage and intersection operations according to

the continuity of transaction data, thereby significantly reducing memory consumption and improving the efficiency of frequent itemset mining. HiAGL-FIM avoids repeated re-encoding in hybrid approaches through one-time adaptive compression modeling based on continuity ratio after a single scan. It also introduces structure-specific intersection operators and a unified support calculation formula for Run–Run, Gap–Gap, and Run–Gap cases, which improve time efficiency and memory compactness under the premise of completely removing bitmap dependency.This article will provide a detailed introduction to the theoretical basis, specific implementation methods, and experimental verification results of the HiAGL-FIM algorithm in subsequent chapters, in order to demonstrate its excellent performance and wide applicability on different types of datasets.

The remainder of this paper is organized as follows: Section II reviews related work on frequent itemset mining and hybrid TID representations. Section III details the proposed HiAGL-FIM algorithm, including its structure, generation process, and complexity analysis. Section IV presents experimental results and performance comparisons with state-of-the-art algorithms. Section V concludes the paper and discusses future research directions.

II. Related Work

Frequent itemset mining algorithms have undergone decades of development, and researchers have proposed a series of classic algorithms and optimization strategies, mainly divided into two categories: horizontal data formats and vertical data formats. The algorithms based on horizontal data formats are represented by the classic Apriori [12] and FP Growth [3]. Among them, the Apriori algorithm first proposed a strategy of generating candidate itemsets layer by layer and pruning them, effectively reducing the search space. However, in large-scale datasets, a large number of candidate sets will be generated, leading to serious performance bottlenecks; FP Growth compresses data through a tree structure (FP Tree), avoiding the generation of candidate sets and improving mining efficiency. However, in the process of building and maintaining FP Tree, it still faces significant memory and time overhead, especially in high-dimensional data and high workload scenarios [13].

The vertical data format algorithms are represented by Eclat [5], Charm [14], and dEklat [15]. The Eclat algorithm represents the transaction support information of each itemset through a transaction identifier list (TID), greatly reducing the number of database scans and suitable for fast mining of large-scale transaction data; Charm further utilizes the characteristics of closed itemsets to optimize the vertical mining process and obtain closed frequent itemsets in a more efficient way; DEklat, on the other hand, reduces the overhead of intersection operations between TID lists through an extended Diffset structure, significantly improving mining efficiency [16]. However, these classic vertical formatting methods all default to using a single type of data structure to represent transaction support information, making it difficult to simultaneously meet performance requirements under different data densities.

To address this issue, researchers have recently begun exploring adaptive data representation structures to adapt to changes in data distribution. Tseng et al. proposed the FPL/TPL framework, which dynamically switches between sparse array and bitmap representations based on the local density of transaction data. Although it improves computational efficiency, the additional overhead caused by data structure transformation operations is particularly prominent in scenarios with frequent changes in data density [7]. Similarly, the NegNodeset+N-list hybrid storage method proposed by Poovan et al. also faces the burden of structural transformation in density changing scenarios, especially with poor performance when the transaction data scale expands [17]. In addition, the HybridMiner algorithm developed by Bashir et al. uses a combination of bitmap and array representations. Although it has improved mining efficiency, it still cannot fundamentally solve the problem of space waste caused by the linear growth of bitmap storage with transaction size [9].

To reduce the high memory requirements of bitmap representation structures, the Ramp algorithm has designed an efficient bit vector projection technique, which to some extent reduces the redundant storage of bitmaps. However, it still generates a large amount of redundant space when transaction data density is sparse [10]. For data stream mining scenarios, the CICLAD algorithm proposed by Martin et al. utilizes compact data structures to improve memory usage efficiency. However, in high-density and volatile scenarios, this algorithm still suffers from reduced computational performance due to frequent data structure adjustments [11]. The existing frequent itemset mining algorithms have more or less the following bottlenecks: 1) a single data representation structure is difficult to accommodate the different density features of transaction data, resulting in insufficient adaptability of the algorithm. 2) Although mixed data representation methods can partially solve adaptability problems, the additional computational burden brought by data structure transformation limits their application effectiveness when transaction density frequently changes. 3) Although traditional bitmap representation is efficient in dense scenarios, it still faces significant memory consumption issues when processing datasets with transaction scales of tens of millions or even billions. Therefore, it is particularly important to design a frequent itemset mining algorithm that does not require bitmap and can automatically adapt to transaction data of different densities. The HiAGL-FIM algorithm proposed in this article adopts a hierarchical adaptive Gap Run transaction identifier representation structure, which completely avoids the shortcomings of bitmap representation, effectively balances the algorithm's space occupation and computational efficiency, and further promotes the practical application of frequent itemset mining in large-scale transaction data.

III. HiAGL-FIM Algorithm

A. Algorithm Framework

The design goal of HiAGL-FIM is to introduce an adaptive transaction identifier compression structure in the modeling stage, combined with structure-specialized intersection operations and layered scheduling mechanisms, to maintain efficiency and stability in large-scale environments with mixed dense and sparse transactions. To achieve this objective, the overall process is divided into five tightly connected phases.

1) Phase 1: Transaction scanning and inverted TID construction: The transaction database $D = \{\tau_1, \ldots, \tau_n\}$ is

scanned linearly to construct the transaction identifier sequence $T_i = \{t_1 < \cdots < t_{|T_i|}\}$ for each item $i \in \mathcal{I}$, while also performing global support counting. This phase completes the initialization of the inverted structure, laying the foundation for subsequent compression and pruning.

2) Phase 2: Continuity-driven adaptive compression: The structural choice is determined by the continuity ratio $\phi(T_i)$:

$$\phi(T_i) = \frac{\sum_{k=1}^{|T_i|-1} \mathbf{1}[\,t_{k+1} - t_k = 1\,]}{|T_i| - 1} \tag{1}$$

When $\phi(T_i) \geq \rho_0$, T_i is compressed into a Run-List; otherwise, it is compressed into a Gap-List. The compression operation is performed once during the modeling phase and remains stable throughout mining, avoiding the overhead of frequent structure switching in mixed-density scenarios.

3) Phase 3: Frequent 1-itemset extraction and buffer partitioning: For each T_i, support is calculated as the sum of segment lengths if stored as a Run-List, or as the element count if stored as a Gap-List. Items with $\sup(i) \geq \theta$ are identified as frequent 1-itemsets and included in the initial result set. Simultaneously, frequent 1-itemsets are partitioned into high-, medium-, and low-frequency buffers, establishing a layered scheduling mechanism to enhance cache locality and prepare for subsequent intersection operations.

4) Phase 4: Structure-specialized intersection and pruning: Candidate itemset expansion adopts a depth-first strategy, and support calculation relies on structure-specialized operators:

$$\mathrm{supp}(A \cap B) = \begin{cases} \sum_{(s,\ell)\in(R_A \cap R_B)} \ell, & \text{Run} \times \text{Run}, \\ |G_A \cap G_B|, & \text{Gap} \times \text{Gap}, \\ |\, G_{\text{gap}} \cap \mathrm{Expand}(R_{\text{run}})\,|, & \text{Run} \times \text{Gap}. \end{cases} \tag{2}$$

Here, $R_\star$ denotes the Run-List and $G_\star$ denotes the Gap-List. During the intersection process, the upper bound of support is used for short-circuit pruning. If the upper bound falls below the threshold θ, the computation is terminated immediately to reduce unnecessary overhead.

5) Phase 5: Recursive expansion and termination: Candidate itemsets that meet the support threshold are recursively expanded, with their TID structures retained as new prefixes. The depth-first expansion strategy ensures a low memory footprint and strong path locality, enabling the algorithm to scale effectively on large datasets. Recursive traversal also allows heuristic pruning methods to be combined, further improving overall execution efficiency.

B. Principle

HiAGL-FIM (Hierarchical Adaptive Gap-Run List-based Frequent Itemset Miner) is an efficient frequent itemset mining algorithm for large-scale transaction data. Its core lies in constructing a hierarchical adaptive transaction identifier representation structure—HAGL-TID (Hierarchical Adaptive Gap-Run Transaction Identifier List). This structure breaks away from the reliance of traditional Eclat and its variants on bitmaps and static sparse arrays. It offers high compression capabilities and density adaptability, effectively improving mining efficiency and resource utilization in both dense and sparse transaction data environments.

The overall process of HIAGL-FIM follows the algorithm framework of "one-time modeling, adaptive compression, structure-specialized intersection, recursive deep expansion". First, the algorithm scans the transaction database $\mathcal{D} = \{t_1, t_2, \ldots, t_n\}$ in sequence, and constructs the tid list T_i for each item $i \in I$. Then, the algorithm measures the local density of the TID list by defining the continuity ratio $\phi(T_i)$ as in Eq. 1. If adjacent TIDs are continuous, the function evaluates to 1. When $\phi(T_i) \geq \rho_0$, the transaction sequence exhibits high continuity. The algorithm compresses it into a *Run-List* structure, recording the starting position and length of each continuous segment; otherwise, it preserves the sequence as an ascending sparse integer array (i.e. a *Gap-List*). This structural compression is completed during the initial database scan and remains static throughout the mining phase, eliminating the re-encoding overhead inherent in traditional dynamic hybrid structures.

To further improve access efficiency, HIAGL-FIM introduces the support-layered buffer mechanism, which divides all TID structures into high-frequency, intermediate-frequency, and low-frequency levels according to their support values, and manages them separately to strengthen memory locality.

During the mining phase, the algorithm uses a depth-first search (DFS) strategy to recursively expand candidate itemsets. In each expansion process, the pre-built inverted TID structure is used to locate the transaction sets between candidate itemsets, and the structure-specialized intersection operator is selected to efficiently calculate the intersection support according to the TID structure combination, including:

- Run × Run: sliding window matching for continuous segments;
- Gap × Gap: sparse array processing with double-pointer or vectorization;
- Run × Gap: fast positioning and screening through interval judgment and mask operations.

When the support of the intersection result is lower than the preset minimum threshold θ, the algorithm prunes immediately to avoid invalid expansion. Through this structure-driven recursive expansion mechanism, HIAGL-FIM maintains stable mining performance in dense and sparse mixed scenarios, while significantly reducing memory consumption and intersection computational complexity. The overall process is shown in Fig. 1.

C. Generation Process

The complete generation process of HIAGL-FIM algorithm includes five stages: transaction preprocessing and inverted structure construction, continuity compression modeling, frequent 1-itemset extraction and buffer partition, candidate itemset intersection support calculation, and recursive expansion to generate frequent itemsets. The process is built around the HAGL-TID structure, forming a seamless mapping from the original transaction to efficient frequent itemsets.

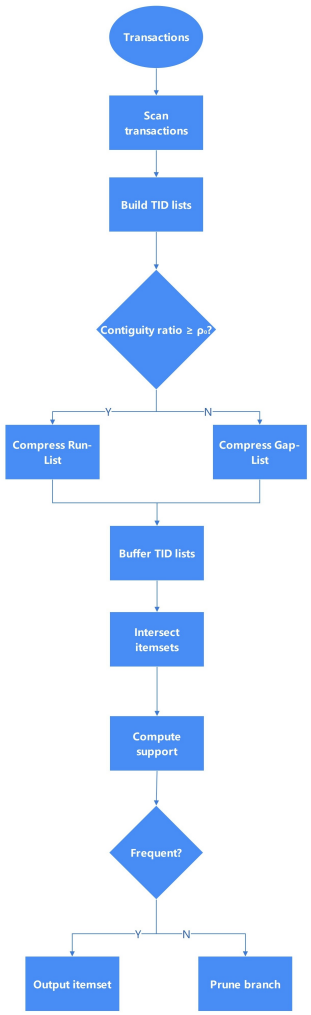


Fig. 1. The overall process of HIAGL-FIM.

1) Transaction scanning and inverted TID structure construction: Firstly, the algorithm performs a single round linear scan on the input transaction database $D = \{\tau_1, \tau_2, \ldots, \tau_n\}$. For each transaction τ_j, write the transaction number j of all items i into its corresponding inverted TID list T_i. At the same time, the algorithm counts the global support of each item to prepare for the subsequent compression modeling and pruning.

2) Modeling of transaction ID compression driven by continuity: After the TID list is built, the algorithm sets the transaction number sequence for each item $T_i = \{t_1, t_2, \ldots, t_{|T_i|}\}$ and conducts continuity assessment by calculating its continuity ratio as in Eq. 1. This ratio describes the intensity of transaction distribution.

According to the continuity threshold ρ_0 (e.g. 0.25), the algorithm determines the structural representation:

- If $\phi(T_i) \geq \rho_0$, it will be compressed into a Run-List structure, recording the starting position and length of each continuous transaction segment.
- Otherwise, the Gap-List structure is retained, representing all TID numbers as an ascending sparse array.

The compression operation is performed only once, and the structure type remains unchanged throughout the mining process. This avoids the problem of heavy recoding of hybrid structures in density fluctuation scenarios, improving stability and execution efficiency.

- Frequent 1-itemset extraction and hierarchical buffer partition: The algorithm then calculates the support of TID structure T_i of each item i:
  - For Run-List: support is the sum of the lengths of all sections.
  - For Gap-List: support is the number of elements in the array.

If support $\sup(i) \geq \theta$ (minimum support threshold), item i is determined to be a frequent 1-itemset. All frequent 1-itemsets are included in the result set and serve as the starting point for subsequent candidate extensions.

At the same time, the algorithm divides its TID structure into high-frequency, intermediate-frequency and low-frequency buffers according to the size of each support, and uses a hierarchical scheduling mechanism to improve access locality and cache hit rate, providing structured support for the intersection calculation stage.

- Intersection calculation and support determination of candidate itemsets

Starting from frequent 1-itemsets, the algorithm uses depth-first search (DFS) to recursively construct candidate itemsets. For two frequent k-itemsets A and B with the same prefix, if their first $k-1$ items are identical, the connection extension is performed to generate the candidate itemset $C = A \cup B$.

To calculate the support of candidate itemset C, the algorithm calls the structure-specific intersection operator, following the unified formula given in Eq. 2.

The formula provides the mathematical basis for support determination and subsequent pruning strategies and recursive expansion. To improve efficiency, the algorithm introduces a short-circuit mechanism during intersection—when the maximum possible support is known to be below θ before completing intersection, the calculation terminates immediately and skips subsequent extensions, effectively reducing unnecessary computational overhead.

- Recursive extension and termination of frequent itemsets.

For each candidate itemset C that passes the support determination, the algorithm recursively expands the next-level candidate set as a new prefix, and continues to perform connection, intersection and pruning until no more itemsets meeting the support requirements can be generated. The depth-first expansion strategy has low memory footprint and path locality, and shows good scalability in large-scale transaction data. In addition, recursive traversal allows the combination of heuristic pruning methods to further improve execution efficiency.

Through the above five stages, HIAGL-FIM not only ensures the high compression rate and structural stability of

the algorithm, but also realizes the efficient recursive mining of frequent itemsets. Especially in the aspect of support calculation, a unified and scalable mining engine is constructed based on the structure-specific intersection strategy of Run-List and Gap-List structures and the formal support determination formula. This algorithm does not need to rely on the bitmap structure, and avoids the frequent switching of hybrid structures, comprehensively improving adaptability in large-scale dense-sparse mixed transaction environments. The pseudo-code is as follows, and experimental verification will be carried out in the following chapters.

Algorithm 1 HiAGL-FIM: Hierarchical Adaptive Gap-Run List Frequent Itemset Miner

Require: Transaction database T, minimum support fraction $minsup_fr$
Ensure: Frequent itemsets F
Load T from file
Initialize *item_to_tidlist* as empty dictionary
for each transaction T_i in T **do**
  for each item x in T_i **do**
    Append i to *item_to_tidlist*[x]
  end for
end for
for each item x in *item_to_tidlist* **do**
  Convert TID list to NumPy array
  if continuity_ratio $\geq \rho_0$ **then**
    Compress to Run-List format
  else
    Compress to Gap-List format
  end if
end for
Compute $minsup \leftarrow \lceil minsup_fr \times |T| \rceil$
Filter itemsets with support $\geq minsup$ into L_1
$F \leftarrow L_1$, $level_k \leftarrow L_1$, $k \leftarrow 1$
while $level_k \neq \emptyset$ **do**
  $next_level \leftarrow \emptyset$
  Group $level_k$ by common $(k-1)$-prefix
  for each group G **do**
    Sort G lexicographically
    for $i \leftarrow 0$ to $|G|-1$ **do**
      for $j \leftarrow i+1$ to $|G|-1$ **do**
        $A \leftarrow G[i]$, $B \leftarrow G[j]$
        $C \leftarrow A \cup B$
        if $C \notin F$ **then**
          Compute $TID(C) \leftarrow TID(A) \cap TID(B)$ using Eq. 2
          if $support(C) \geq minsup$ **then**
            Store $TID(C)$ (auto-adaptive)
            Add C to F and $next_level$
          end if
        end if
      end for
    end for
  end for
  $level_k \leftarrow next_level$
  $k \leftarrow k+1$
end while
return F

D. Structural Optimization and Complexity Analysis

To comprehensively improve the performance of HiAGL-FIM in large-scale transaction data processing, the algorithm introduces various optimization mechanisms in the structure design and execution process. This section systematically describes these optimization strategies, analyzes their time and space complexity across different modules, and highlights the algorithm's scalability and practicability.

1) Compression coding and pruning acceleration mechanism: HiAGL-FIM compresses all TID lists at once through the continuity ratio $\phi(T_i)$ as defined in Eq. 1 during the modeling stage, avoiding the recoding overhead caused by frequent hybrid structure switching in traditional methods. The compression process has linear time complexity $O(n)$, where n is the number of TIDs.

During candidate itemset expansion, the algorithm introduces support upper-bound estimation combined with structure-specialized intersection operators (Eq. 2) to enable "intersection short-circuit" judgment. If the current intersection cannot reach the minimum support threshold θ, immediate pruning avoids subsequent invalid calculations.

2) Buffer hierarchical scheduling: TID structures with different support levels are allocated to high-frequency, intermediate-frequency, and low-frequency buffers for batch management and cache-affinity scheduling. This mechanism significantly reduces random TID structure accesses during intersection operations, improving memory locality and reducing access latency.

- Time and space complexity analysis
  - Initialization phase (build TID table): Single transaction scan with time complexity $O(n \cdot L)$ and space complexity $O(n)$, where n is the transaction count, L is the average transaction length, and m is the number of distinct items.
  - Structure compression stage: Calculate $\phi(T_i)$ for each item and classify structures. Complexity $O(n \cdot m)$, where m is the average support.
  - Intersection calculation phase:
    - `Run × Run`: optimal complexity $O(p+q)$ with p, q segment counts;
    - `Gap × Gap`: complexity $O(m+n)$, reducible to $O(\min(m,n))$ with hash optimization;
    - `Run × Gap`: approximately $O(n \log p)$ using binary/mask matching.
  - Recursive expansion stage: DFS approach prevents candidate explosion. Actual depth depends on frequent itemset density. Worst-case remains exponential $O(2^n)$, but pruning mechanisms effectively control expansion scale.

IV. Result and Discussion

All experiments were conducted in the same computing environment. The experimental platform was a personal computer equipped with 2.2 GHz Intel Core i5 processor, 8 GB memory and 64-bit Windows 10 Professional operating system. To ensure comparability and consistency of results, this paper implements all algorithms using Python without loading any GPU acceleration components during experiments, guaranteeing authenticity and reliability of measured performance metrics.

For experimental validation, we selected the classic sparse transaction database `T10I4D100K` as test dataset, containing 100,000 transactions with an item set size of 870. Each transaction averages approximately 10 items, exhibiting typical low-density characteristics. This dataset is widely adopted in frequent itemset mining research and particularly suitable for evaluating algorithm efficiency and resource consumption in sparse data scenarios.

To comprehensively assess HiAGL-FIM's performance advantages, we compare against two mainstream algorithms: FP-Growth and Eclat. FP-Growth avoids explicit candidate generation by constructing compressed FP-Tree structures, representing one of the most widely-used horizontal format algorithms. Eclat employs vertical TID representation and computes support through intersection operations, offering good execution efficiency but with performance highly dependent on underlying TID structure design. For fair comparison, all algorithms run independently under identical input data and parameter configurations, with key performance metrics recorded at five different support thresholds.

In this paper, the support threshold σ is set as $\{0.006, 0.005, 0.004, 0.003, 0.002\}$ to simulate the actual mining demand change from high support to low support. Under each support setting, the running time (in seconds) and the peak physical memory usage (in MB) required by the algorithm to complete the mining of all frequent itemsets are measured respectively. The memory statistics are collected in real time through the psutil library. The evaluation results are shown in Fig. 2 and Fig. 3, respectively corresponding to the running time change and memory consumption trend under different support levels.

As can be seen from Fig. 2, as the support threshold σ decreases from 0.006 to 0.002, the running time of *HiAGL-FIM* gradually increases from 10.9 seconds to 18.7 seconds. Although the computational workload increases due to the exponential expansion of candidate itemsets at a lower threshold, *HiAGL-FIM* consistently outperforms Eclat and FP-growth across the entire threshold range. Specifically, the average acceleration of *HiAGL-FIM* is 31.7% over Eclat and 45.9% over FP-growth, demonstrating its superior time efficiency. This performance advantage stems from its layered compression mechanism, which enables fast and structure-aware TID intersection.

Fig. 3 further highlights the significant memory-saving advantage of the Hierarchical Adaptive Gap-Run List (HAGL). The memory footprint of FP-growth and Eclat remains relatively high—ranging from 383.58 MB to 421.43 MB and 136.85 MB to 142.02 MB respectively—while the memory usage of *HiAGL-FIM* remains compact, ranging only from 4.76 MB to 13.15 MB. At the lowest threshold $\sigma = 0.002$, the memory consumption

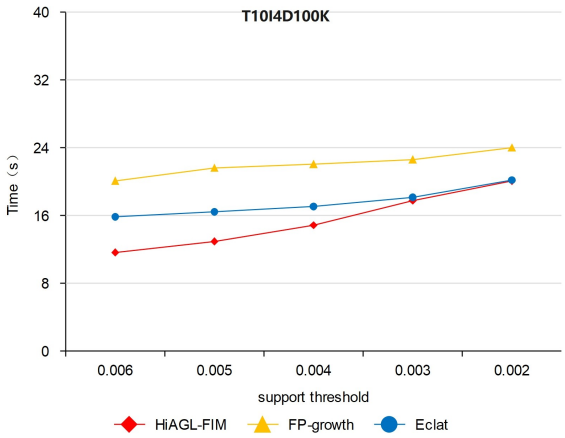


Fig. 2. Execution time comparison under different support thresholds on T10I4D100K.

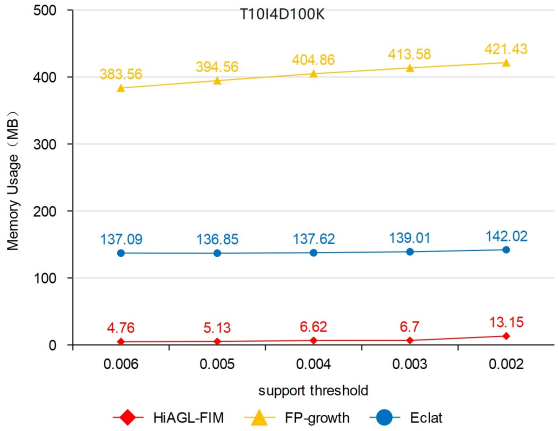


Fig. 3. Memory usage comparison under different support thresholds on T10I4D100K.

of *HiAGL-FIM* is just 13.15 MB, accounting for only 9.2% of Eclat and 3.1% of FP-growth.

This substantial improvement is attributed to the use of the HAGL-TID representation, which dynamically adapts to the density of TID sequences by switching between compact Run-List encoding for continuous intervals and sparse Gap-List encoding for discrete elements. In addition, the algorithm employs specialized intersection operators for each structure pair (Run-Run, Gap-Gap, Run-Gap), thereby minimizing redundant operations in the candidate generation process. As a result, the overall space complexity is effectively reduced to $\mathcal{O}(|G| + |R|)$, where $|G|$ and $|R|$ represent the size of the Gap and Run segments, respectively. This lightweight design enables *HiAGL-FIM* to efficiently mine frequent itemsets from large-scale transaction datasets in a single-threaded Python environment.

V. Conclusion and Future Work

In this paper, a new frequent itemset mining algorithm, HiAGL-FIM, is proposed. Its core lies in the design of a Hierarchical Adaptive Gap-Run TID (HAGL-TID) structure.

Through continuity-driven transaction compression modeling, structure-specialized intersection operations, and hierarchical buffer scheduling mechanisms, the algorithm achieves efficient processing of dense–sparse mixed environments in large-scale transaction databases. Experimental results demonstrate that, compared with the classical Eclat and its variants, HiAGL-FIM exhibits significant advantages in execution time and memory consumption, verifying the effectiveness and scalability of the proposed method in large-scale data scenarios.

However, the proposed method still has certain limitations. The current study mainly focuses on validation using static transaction databases and does not fully address dynamic updates in streaming data environments. In addition, the implementation is based on a single-machine architecture without distributed or parallel optimization, leaving room for improvement in handling ultra-large-scale datasets.

Future research can be carried out in the following directions: 1) extending HiAGL-FIM to streaming environments by incorporating sliding windows and incremental update strategies to achieve efficient mining of dynamic data; 2) introducing distributed computing frameworks and GPU acceleration to enhance the algorithm's processing capacity in large-scale scenarios; and 3) exploring the integration of frequent itemset mining with concept drift detection, pattern evolution, and other tasks, further expanding the application potential of the algorithm in intelligent decision-making and real-time data analysis.

REFERENCES

[1] W. I. D. Mining, *Data Mining: Concepts and Techniques*. San Francisco, CA: Morgan Kaufmann, 2006.

[2] C. C. Aggarwal, *Data Mining: The Textbook*, vol. 1. New York: Springer, 2015.

[3] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *ACM SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, 2000.

[4] G. Grahne and J. Zhu, "Efficiently using prefix-trees in mining frequent itemsets," in *Proc. FIMI*, 2003, p. 65.

[5] M. J. Zaki, "Scalable algorithms for association mining," *IEEE Trans. Knowl. Data Eng.*, vol. 12, no. 3, pp. 372–390, 2000.

[6] M. R. Al-Bana, M. S. Farhan, and N. A. Othman, "An efficient spark-based hybrid frequent itemset mining algorithm for big data," *Data*, vol. 7, no. 1, p. 11, 2022.

[7] F. C. Tseng, "An adaptive approach to mining frequent itemsets efficiently," *Expert Syst. Appl.*, vol. 39, no. 18, pp. 13,166–13,172, 2012.

[8] J. S. P. Poovan, D. A. Udupi, and N. V. S. Reddy, "A multithreaded hybrid framework for mining frequent itemsets," *Int. J. Elect. Comput. Eng.*, vol. 12, no. 3, pp. 3249–3264, 2022.

[9] S. Bashir and A. R. Baig, "HybridMiner: Mining maximal frequent itemsets using hybrid database representation approach," in *Proc. INMIC*, 2005, pp. 1–7.

[10] S. Bashir and A. R. Baig, "Ramp: Fast frequent itemset mining with efficient bit-vector projection technique," *arXiv:0904.3316*, 2009.

[11] T. Martin, G. Francoeur, and P. Valtchev, "CICLAD: A fast and memory-efficient closed itemset miner for streams," in *Proc. KDD*, 2020, pp. 1810–1818.

[12] L. Huang, H. Chen, X. Wang, and G. Chen, "A fast algorithm for mining association rules," *J. Comput. Sci. Technol.*, vol. 15, pp. 619–624, 2000.

[13] J. Lepping, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. Hoboken, NJ: Wiley, 2018.

[14] M. J. Zaki and C. J. Hsiao, "CHARM: An efficient algorithm for closed itemset mining," in *Proc. SDM*, 2002, pp. 457–473.

[15] M. J. Zaki and K. Gouda, "Fast vertical mining using diffsets," in *Proc. KDD*, 2003, pp. 326–335.

[16] B. Vo, T. Le, F. Coenen, and T. P. Hong, "Mining frequent itemsets using the N-list and subsume concepts," *Int. J. Mach. Learn. Cybern.*, vol. 7, pp. 253–265, 2016.

[17] B. Wang, X. X. Fang, R. R. Lv, and J. J. Ma, "Weighted frequent itemset mining algorithm based on WNegNodeset structure," *J. Comput. Appl.*, vol. 37, no. 7, 2020.

Optimizing Energy Efficiency and Increasing Scalability in 6G-IoT Networks Through SDN, Duty Cycling, and AI-Driven Slicing

Marwah Albeladi, Kamal Jambi, Fathy E. Eassa, Maher Khemakhem
Department of Computer Science-Faculty of Computing and Information Technology,
King Abdulaziz University (KAU), Jeddah 21589, Saudi Arabia

***Abstract*—As sixth-generation (6G) and Internet of Things (IoT) networks expand rapidly, concerns are growing about their energy consumption and scalability. This is primarily because more devices are being connected, resulting in increased energy consumption.energy consumption.This study examines three primary strategies for optimizing energy efficiency and improving scalability in 6G-IoT networks.This research looks at three experimental setups: 1) using software-defined networking (SDN) with dynamic slicing to organize devices based on when they are most and least used, 2) duty cycling, which turns devices on and off to save energy, and 3) AI-optimized network slicing that uses both convolutional neural networks (CNN) and bidirectional long short-term memory (BiLSTM) models. In the first setup, SDN with dynamic slicing helped reduce unnecessary power consumption by matching device activity to peak times. As more devices were added, this method kept energy use low and improved the network's ability to handle growth without requiring significantly more power. This resulted in a 66.28 percent decrease in power usage. In the second setup, duty cycling allowed only some devices to be active at a time, which reduced power use by over 60 percent during slow periods. In the third setup, the CNN-BiLSTM model effectively classified service types and reduced power use by 60.14 percent. While these methods were not combined into a single solution, each utilized slicing techniques to more effectively allocate resources and manage power.**

Keywords*—6G-IoT; energy efficiency; scalability; SDN; duty cycling; network slicing; CNN; BiLSTM; AI-driven optimization

I. Introduction

This study addresses the urgent need for smart, energy-efficient slice management solutions in 6G networks, targeting the challenge of supporting advanced AI services with new QoS requirements. By developing innovative management strategies, the research contributes to improving energy efficiency, reliability, and adaptability in next-generation SAGIN-enabled 6G-IoT networks [1] [2]. Many technologies, including mobile phones, transportation systems, food production, housing, healthcare, clothing, and remote monitoring, are being transformed by the Internet of Things (IoT). As these advances occur, creating energy-efficient 6G-IoT networks becomes crucials [3].

Existing studies have shown that 6G-IoT networks will link billions of devices worldwide, resulting in high energy consumption. While prior research highlights the significance of energy efficiency for the long-term profitability of network operators, the current study distinguishes itself by focusing on the integration of space, air, and ground networks into a Space-Air-Ground Integrated Network (SAGIN) [4].

Unlike earlier works centered on terrestrial networks, this research emphasizes global coverage and intelligent management of network slices to ensure specific Quality of Service (QoS) needs [5]. Network slicing, as described in prior studies, enables logically isolated virtual networks for different services; this research extends the discussion by analyzing cost-effective slice management strategies throughout the slicing life cycle, which past research has only partially addressed.

Unlike earlier generations, which typically managed network resources within a single domain, 6G network slicing must coordinate across heterogeneous segments in SAGIN. Previous studies have not fully considered the additional complexity that emerges from integrating space, air, and ground networks. The current research addresses this gap by focusing on unique challenges and management strategies necessary for effective slice orchestration across these domains.

Therefore, the development of smart slice management solutions in 6G networks is required.

The current research addresses this need by proposing an approach for efficient and intelligent network slice management specific to SAGIN-based 6G-IoT environments. This work aims to advance current practice by focusing on enhanced resource allocation and tailored slice management strategies that specifically address the emerging challenges identified above. In addition, with the availability of powerful computing capabilities and advanced.

AI services are being enhanced with new QoS requirements, such as data quality, training latency, and inference accuracy. As a result, dedicated network slices must be established to support emerging AI services in 6G networks. To address these challenges, innovative solutions are urgently needed to enhance energy efficiency without compromising network performance [6]. The objective of this paper is to:

- Analyzing the impact of SDN on energy consumption in 6G-IoT networks, by evaluating the dynamic allocation mechanism for devices based on peak and idle periods.
- Evaluating the effectiveness of Duty Cycling in reducing unnecessary energy consumption by regulating device on- and off-duty times based on daily work schedules.

- Testing the performance of a deep learning model consisting of CNNs and BiLSTM networks in classifying network data into multiple categories and accurately estimating the energy requirements for each category.
- Identifying the energy efficiency differences between each technology separately without combining them, to obtain a clear and independent picture of the impact of each approach under different operating conditions.
- Providing a scientific basis for comparing energy management methodologies to enable 6G-IoT network developers to choose or develop flexible and scalable solutions based on the nature of network load and future applications.

This study presents a set of contributions used to reduce energy consumption and increase efficiency in the 6G-IoT network through the following points.

- SDN-based dynamic slicing was combined into one model to reduce energy consumption and increase energy efficiency.
- Slicing and duty cycling to reduce energy consumption and increase energy efficiency.
- Explain how energy efficiency changes with increasing device density, ensuring the ability to adapt to network expansion processes.
- hybrid CNN-BiLSTM model was implemented, achieving 99% classification accuracy, enabling intelligent slicing and adaptation.

This paper is organized as follows. To begin, Section II covers the key features of 6G and the role of AI-powered network chips, laying the groundwork for the subsequent discussion. Building on this, Section III reviews previous work on energy efficiency and scalability in 6G IoT. Then, Section IV outlines our research approaches, which are expanded upon in Section V with details on the proposed approaches, including SDN-based chips, duty cycle, and AI-enabled chips. Following that, Section VI reviews the results and effectiveness of these techniques. In Section VII, we further discuss our experimental results on energy usage. Finally, Section VIII concludes the paper and suggests future research directions.

II. Background

Network slicing is a technology that divides a network into multiple dedicated "slices". Each slice operates independently, allowing flexible allocation of resources based on specific needs. This technology significantly enhances network performance and quality, particularly in expanding IoT environments [7]. With network slicing, resources can be given to each slice based on its unique needs.

For example, one slice can serve IoT applications that require steady connectivity and low power, while another can support high-speed data, such as HD video. Allocating resources based on actual service or application needs reduces waste and improves network efficiency [9]. Each slice works independently, reducing interference between applications. 6G networks can support a wide range of diverse applications, from self-driving cars to virtual and augmented reality services. Each of these applications requires specific network characteristics, such as instant response or high bandwidth. With network slicing, these specific needs of each application can be met individually. In addition, robust security strategies can be applied to each slice to ensure data protection and provide faster response.

Machine learning can monitor network activity and predict the needs of devices and services [10]. It can analyze large datasets to provide the optimal allocation of resources. In addition, it helps reduce network congestion by optimizing data distribution. Machine learning can be used to improve services such as security, spam detection, and power management automatically. 6G networks will provide a range of different services that will benefit users. These services include enhanced mobile broadband, ultra-reliable low-latency communication, and massive machine-type communications [11]:

- Super-eMBB: means broadband connectivity via mobile phones, with a focus on energy efficiency.
- Massive MTC: means connecting a very large number of devices, such as IoT devices, that need constant connectivity.
- Super URLLC: provides highly reliable communications with minimal delay, such as remote control of devices.
- Ultra-high resolution: means using technologies to provide high resolution in data transmission.
- Super-immersive reality: includes virtual reality and augmented reality experiences that provide enhanced interaction.

III. Related Work

Sixth-generation - 6G networks rely on network slicing, a technology that is still in its infancy but is rapidly evolving and offering a variety of services.

A. Using Software-Defined Networking

The study [16] used the energy-aware routing, multi-level, and mapping problem (EARMLP) algorithm, which achieves better performance by reducing the number of active nodes and integrating the use of network resources. The number of controllers and their optimal placement have a significant impact on energy savings. In [17], the authors present a comprehensive survey on SDN for various smart applications. This survey covers the infrastructural details of SDN hardware, OpenFlow switches, controllers, simulation tools, programming languages, open issues, and challenges in SDN implementation using advanced technologies.

B. Using Machine Learning

Several studies have used different methods to reduce power consumption. In a study [18], collaborative communication was used. This means that mobile phones or smart devices work together as a team, rather than working separately. When these devices work together, energy can be more efficiently saved, which helps extend battery life. Machine

learning, specifically artificial neural networks (ANNs), is utilized to enhance network slicing in 6G networks, focusing on energy consumption. This approach involves a multifaceted strategy that integrates various techniques to improve energy efficiency while maintaining high performance. In [19], the data rate allocation (DTRA) method was used to improve data transmission efficiency in 6G networks. The residual energy cluster head (RECH) method was used. Furthermore, this work used the dynamic multipath routing protocol (DMRP)method to improve the reliability and speed of 6G networks. After evaluating performance metrics, the DTRA method improved the lifetime and energy efficiency of the network by 95.3% based on 6G networks.

C. Using Network Slicing

Flexible network slicing is one of the essential components of 6G networks, allowing the creation of customized network environments to meet the needs of specific applications and services. The study of Sheena [27] aims to improve the efficiency of the network by designing a Deep Learning-based Network Slicing with Data Aggregation (EENS-DA) technique, which allocates the necessary physical resources to specific applications clearly and efficiently. The study of Phyu [28]aims to address the problem of activating/ deactivating slicing to reduce energy consumption while maintaining Quality of Service (QoS) for users. The researchers relied on two Multi-Armed Bandit (MAB) agents to make activating/deactivating decisions at the level of individual base stations. Researchers in the study [31] propose a hybrid model that combines CNN and BiLSTM. The CNN was used to extract automatic features from the input data. The BiLSTM was used to classify and determine the appropriate network segment for each request. The results showed that the hybrid model achieved an overall accuracy rate of 97.21%, demonstrating the effectiveness of this approach in allocating the appropriate network segments to end users.

D. Research Gap

After reviewing existing research, several gaps in the literature become apparent. Some studies focus on specific techniques, such as network slicing, machine learning, and energy-aware routing. However, a comprehensive framework that integrates these methods to improve energy efficiency in 6G-IoT networks remains absent. Although energy efficiency has generally improved, challenges persist in maintaining quality of service (QoS), which includes factors such as latency and throughput. Only a limited number of studies have successfully addressed the simultaneous enhancement of both energy efficiency and QoS, as demonstrated in the work by [28]. Flexible network slicing is widely recognized as a crucial component for the success of 6G; nevertheless, established guidelines for the design, implementation, and management of such slices are lacking, as noted in studies [27] and [28]. Addressing these gaps is expected to contribute to significant advancements in energy efficiency and scalability in 6G-IoT networks through the use of advanced technologies such as machine learning, network slicing, and SDN. Table I provides a comparison of the relevant studies.

IV. Study Contributions

The current study significantly expanded the scope of previous research by simultaneously combining three independent and integrated technologies: software-defined networking (SDN)-based dynamic slicing, workflow, and AI-assisted slicing using CNN-BiLSTM, to improve energy utilization and scalability in 6G-IoT environments. Although each previous study focused on a single aspect, such as software-defined networking (SDN) control plane (Study[16]),or an artificial neural network (ANN)-based collaboration (Study [18]), These efforts often faced critical limitations, including high complexity, poor scalability, and limited accuracy in traffic management.

The current study addresses the gaps in previous research through a diverse experimental design. In the first scenario, the study used Software-Defined Networks (SDN) to dynamically operate devices during peak usage periods. This technology addressed the limitations of static routing in previous studies based on SDN. By applying the second scenario, the study was able to improve selective activation strategies, such as those found in studies [18] and [19]. A duty cycle was implemented in the third scenario to reduce energy consumption during periods of low activity, which is considered the most innovative.

It utilized CNN-BiLSTM technology to intelligently classify traffic types and allocate resources accordingly, surpassing previous models like random forests or DRL in terms of accuracy and energy saving. Table II presents a comparative analysis of related studies and the contribution of the current research.

The current study significantly expanded the scope of previous research by simultaneously combining three independent and integrated technologies: software-defined networking (SDN)-based dynamic slicing, workflow, and AI-assisted slicing using CNN-BiLSTM, to improve energy utilization and scalability in 6G-IoT environments. Although each previous study focused on a single aspect, such as software-defined networking (SDN) control plane (Study [16]), or artificial neural network (ANN)-based collaboration (Study [18]), these efforts often faced critical limitations: high complexity, poor scalability, or limited accuracy in traffic management.

The current study addresses the gaps in previous research through a diverse experimental design. In the first scenario, the study used Software-Defined Networks (SDN) to dynamically operate devices during peak usage periods. This technology addressed the limitations of static routing in previous studies based on SDN. By applying the second scenario, the study was able to improve selective activation strategies, such as those found in studies [18] and [19]. A duty cycle was implemented to reduce energy consumption during periods of low activity in the third scenario, which is considered the most innovative. It utilized CNN-BiLSTM technology to intelligently classify traffic types and allocate resources accordingly, surpassing previous models like random forests or DRL in terms of accuracy and energy saving. Table II presents a comparative analysis of related studies and the contribution of the current research.

TABLE I. COMPARATIVE ANALYSIS OF RESEARCH ON ENERGY EFFICIENCY IN 6G NETWORKS

Author	Idea	Methodology	Features	Challenges
[16]	Energy-aware routing to reduce active nodes and improve resource utilization	EARMLP (Energy-Aware Routing Multi-Level and Mapping Problem)	Reduces number of active nodes; integrates network resource usage	Impact of number and optimal placement of controllers on energy savings
[17]	Survey on SDN for smart applications	Comprehensive survey	Covers SDN hardware, OpenFlow switches, controllers, simulation tools, programming languages; open issues and challenges	Implementation challenges of SDN with advanced technologies
[18]	Reducing power consumption through collaborative communication	Collaborative communication + ANN	Smart devices work together as a team; saves energy; extends battery life	Coordinating collaboration while maintaining high performance
[19]	Improving energy efficiency and lifetime of 6G networks	DTRA; RECH; DMRP	Improved transmission efficiency, reliability and speed; network lifetime/energy efficiency improved by 95.3%	Managing multiple methods together in real 6G environments
[27]	Efficient network slicing with deep learning and data aggregation	EENS-DA (Deep Learning + Data Aggregation)	Allocates physical resources clearly and efficiently to specific applications	Balancing energy saving with application needs
[28]	Energy saving by activating/deactivating slicing while keeping QoS	Multi-Armed Bandit (MAB) agents	Activate/deactivate decisions at base-station level	Maintaining QoS while reducing energy consumption
[31]	Hybrid model for network-slicing classification	Hybrid CNN + BiLSTM	CNN for feature extraction; BiLSTM for classification; overall accuracy 97.21%	Complex training and computational resources

V. METHODOLOGY

In this section, deep learning methods are introduced for tackling the problem of network slicing.

A. Convolutional Neural Network

1) Reasons to use CNN: CNNs [8] can find hidden patterns in data without needing any manual adjustments. High Efficiency: The convolutional layer focuses on specific parts of the data, making it highly effective for analyzing data related to networks. CNNs [12] can create systems that communicate using. Human speech [32]. Making Overprocessing Simpler: Pooling simplifies the data while retaining most of the essential information. CNNs consist of several layers connected in sequence: the first layer is the input layer, followed by hidden layers, and the last one is the output layer. The hidden layers process the input data and extract important features using filters. Overall, the combination of convolutional layers, pooling layers, and fully connected layers in CNNs enables the network to learn and recognize patterns effectively in complex data [33].

B. Long Short Term Memory

1) Reasons to use LSTM : [13]: Long Short-Term Memory Reasons to use LSTM are that it is a unique type of recurrent neural network (RNN) [20] [21] specifically designed to address issues such as vanishing and exploding gradients. These problems can make it challenging for neural networks to learn effectively. For network slicing classification, a deep learning [15]approach was applied. LSTMs [22]are better suited for this task because they can handle both the fading and growing gradient issues, as well as the long-term dependency challenges [14] that regular RNNs face. This makes LSTMs typically more effective than traditional RNNs[33].

C. Bidirectional LSTM (BiLSTM)

1) Reasons to use bidirectional LSTM: Bidirectional LSTM (BiLSTM):Reasons to use Bidirectional LSTM (BiLSTM) [15] include adding an LSTM layer that processes information in both forward and reverse orders. The outputs from the two LSTM [14]layers are then combined using techniques such as calculating the mean, sum, multiplication, or concatenation. The main advantage of using BiLSTM is that it allows each part of the input data to include information from both past and present contexts. This results in more accurate output because BiLSTM [22] uses LSTM layers to analyze data from both directions. Although BiLSTM might seem complicated, it produces strong results due to a solid understanding of the data environment. In [34], a multilayer BiLSTM is utilized, where each layer consists of two cells that process information in forward and backward directions separately.

D. Dataset Description

In Case 3, the dataset is sourced from the University of California, Irvine (UCI) Machine Learning Repository [35]. It includes 87 features, each representing details of an IP flow from a network device, such as source and destination IP addresses, port numbers, and connection timestamps. One source collects this information, while another classifies the layer 7 protocol, which corresponds to the application level in network communication [36]. Most features are numerical, with some being categorical (nominal), and one feature captures dates derived from timestamps [37].

E. Data Preprocessing

The diverse data (textual and numerical) were processed to address value errors, gaps, and duplicates in accordance with the model's requirements. Python was used to remove duplicate rows, handle missing values with the column mean, detect and delete errors, then Min-Max normalization was applied to constrain the values between 0 and 1, and the cleaned data was saved in a new CSV file in preparation for machine learning models. The set includes 78 labels, and the labeling process is a pivotal step directed by device requests and scientific literature [37]. The data was divided into five slices: Super-eMBB, Massive MTC, super-URLLC, super-Precision, and super-immersive; these are common categories in 5G/6G research that represent different service requirements, including high speed, low latency, support for a large number of devices, precision, and reliability.

F. Dataset Segmentation

The data is categorized into the following types:

TABLE II. COMPARATIVE ANALYSIS OF RELATED STUDIES AND THE CONTRIBUTION OF THE CURRENT RESEARCH

Field	Studies (Ref.)	Methodology Used	Objective	Strengths	Gaps	How the Current Study Fills the Gaps
ML for Energy Saving	[18], [19]	Collaborative communication; ANN; DTRA, RECH, DMRP	Reduce power consumption and enhance network efficiency/lifetime	Team-based device operation to save energy and extend battery life; improved transmission efficiency and reliability; network lifetime/energy efficiency improved by 95.3%	No explicit SDN integration or traffic-type slicing; limited unified evaluation across scenarios	Integrates ML within an SDN environment with 6G traffic classification and end-to-end energy impact assessment
Slicing	[27], [28], [31]	EENS-DA (Deep Learning + Data Aggregation); two MAB agents for slice on/off; Hybrid CNN+BiLSTM for slice classification	Efficient resource allocation and maintaining QoS while enabling accurate slice selection	Clear and effective allocation to applications; energy reduction via controlled activation/deactivation; 97.21% overall accuracy for classification	Limited/implicit SDN-based orchestration and unified energy evaluation	Uses SDN with CNN–BiLSTM–based traffic classification to allocate resources by data type and quantify energy gains
SDN	[16], [17]	EARMLP algorithm (energy-aware routing, multi-level mapping); comprehensive SDN survey	Enhance network efficiency and reduce energy consumption via resource-aware routing	Reduced number of active nodes; integration of network resources; broad coverage (hardware, controllers, tools, languages, challenges)	Lack of integration with ML-based classification and 6G slicing context	Combines SDN resource control with ML-driven traffic classification and slicing for holistic energy-efficient 6G management

1) Super-eMBB: This chip is designed to deliver high data speeds and large transfer capacities. It is ideal for applications that need fast data transfers, such as streaming 4K/8K videos and online gaming [38].

2) Massive-MTC: This chip is intended for Internet of Things (IoT) applications, which involve communication among many devices. Examples include smart meters, wearable gadgets, and embedded systems. It excels at managing numerous low-power devices [39] [40].

3) Super-URLLC: This chip is used for applications requiring both high reliability and minimal delay, like self-driving cars and telemedicine. It ensures dependable data transfer while keeping delays to a minimum [41] [42].

4) Super-precision: This chip caters to applications needing high spatial resolution or detailed data, such as environmental monitoring, precise measurements, and augmented reality (AR) and virtual reality (VR) [43].

5) Super-immersive: This chip is designed for applications requiring extensive coverage and high efficiency, such as augmented reality (AR) and virtual reality (VR). It facilitates immersive user experiences within advanced network environments [44].

The categorization process is as follows: when a device, such as a smartphone or sensor, sends a request, the type of application or service is evaluated. Based on this evaluation, the appropriate category, or "slice", is assigned to meet those needs. For example, if high speed is necessary, the request will be routed to the Super-eMBB slice.

In contrast, if minimal delay is essential, as in the case of self-driving cars, the Super-URLLC slice will be employed. Each slice provides distinct performance and resources customized to the current network demands, optimizing resource utilization and minimizing energy consumption.

VI. PROPOSED METHODOLOGY

The approach involves three key experiments designed to address existing research gaps in energy efficiency and scalability of 6G-IoT networks. Each of these experiments employs various methods to reduce energy use in 6G-IoT networks.

A. Case 1 Software Defined Networking with Dynamic Slicing (Scenario 1)

Step 1: The experiment presents a simulation model of a 6G-IoT network in Python using NetworkX [25], starting with 30 devices and gradually expanding to 500-2500 devices to measure scalability and its impact on performance and energy consumption. The network is managed via SDN with dynamic slicing, which adjusts the states of devices (active/inactive) according to peak (6 AM–6 PM) and idle periods. Sensors remain active during peak times, cameras are disabled outside of these periods, and lights operate randomly. The setup relies on networkx, numpy [22], matplotlib [24], and random, and energy is calculated using a function that aggregates the consumption of active devices only, then compares the values before and after slicing. Hourly readings were collected over 24 hours and analyzed using the mean and standard deviation

to estimate efficiency gains and assess the contribution of SDN in optimizing consumption and enabling adaptive device management.

Step 2: The code was improved by introducing a new idea to track power consumption in the "No Slicing" state. A baseline was established for the No Slicing case, recording power consumption for each device category with fixed reference values: sensors (10 units), cameras (8 units), and lighting (6 units). Representative device values (10 sensors, 8 cameras, 6 lighting units) were selected based on real-world datasheets such as the NXP SLN-VIZN-IoT platform ($\approx$ 0.9W) when the camera is on) and the LSM6DSV16X sensor from STMicroelectronics, as well as low-power optical sensors (193–277 µW). This distribution is further supported by the results of the paper [26], which provided practical measurements of power consumption in real industrial environments. These values were developed to standardize comparison conditions and highlight the impact of energy optimization techniques, particularly duty cycle and grid slicing [45].

The results of this case are stored in the no-slicing energy history list and are used with different network sizes to test robustness and allow subsequent replacement with real data without changing the methodology. When applying dynamic slicing within an SDN environment, power is allocated according to peak periods; specific categories (such as sensors) remain active while others are disabled during idle times, and power is measured using the calculate energy function. In this case, the readings are stored in the slicing energy history list, and a direct comparison is made between the no-slicing energy history and slicing energy history lists to measure efficiency gains and estimate the actual reduction in consumption.

Step 3: The simulation of the 6G-IoT network was expanded with SDN management to cover multiple slicing use cases across three device categories (50, 500, and 2,500 devices). Slicing divides the network into smaller, service-oriented segments, reducing energy consumption by distributing loads more efficiently. For each category, energy consumption is measured in both the non-partitioned and partitioned states, and then the percentage reduction attributed to the application of partitioning is calculated.

B. Case 2 Duty Cycling (Scenario 2)

Duty Cycling is applied as a time-based control mechanism to turn devices on and off with the aim of reducing overall energy consumption. In the experimental design, the devices are activated during even hours and deactivated during odd hours, and the consumption of each device is calculated according to its operational state and type. A fixed reference power is used to represent the nominal consumption during continuous operation: sensors 10 watts, cameras 8 watts, and lighting 6 watts. The SDN controller holds the control layer. It activates/deactivates devices according to the duty cycle schedule, ensuring consistent transitions between the two states (active/inactive) at the network level. A baseline without duty cycling is defined where devices are considered always active, Energy is calculated hour by hour as the sum of the consumption of active devices only, allowing the isolation of the impact of scheduling from other factors.

For evaluation purposes, measurements are collected hourly and stored in two separate lists: one for the baseline (no duty cycling energy history) and another for the scheduled case (duty cycling energy history). A direct comparison is made between the two lists to derive the reduction ratio in consumption, with a stratified analysis comparing performance during peak hours versus quiet periods to measure the system's response to load changes. This procedure provides a clear, systematic description: defining reference capabilities for each device category, enforcing scheduling via SDN, consistent hourly measurement, and then an organized before/after comparison, which allows for a more accurate estimation of the efficiency gains resulting from managing on/off states according to a duty cycling schedule.

C. Case3 CNN + BiLSTM (Scenario 3) Model Training

A new AI-driven network slicing model (Scenario 3) has been developed to dynamically improve resource allocation. It relies on a hybrid architecture that combines (CNN for feature extraction) and (BiLSTM for capturing temporal dependencies), classifying network flows into five categories: (super-eMBB, massive-MTC, super-URLLC, ultra-precise applications, and ultra-immersive experiences). The Unicauca IP Flow dataset was used with an 80testing split, and performance was measured using Precision, Recall, and F1 metrics. The classification outputs guide the decision on slice assignment and resource allocation, and then the energy consumed before and after allocation is compared; this demonstrates a better alignment between network traffic requirements, reduced energy consumption, and support for efficient slice management in 6G environments. The study examined the scalability by increasing the number of devices between 500 and 2500 devices, measuring the power consumption of each device as the size changed, and analyzing the relationship between the number of devices and energy efficiency. The features of CNN and BiLSTM were combined through a concatenation layer, and the model was trained on 80% of the data and evaluated on the remaining portion; the results indicate an improvement in efficiency after model-based optimization, as shown in Fig. 1

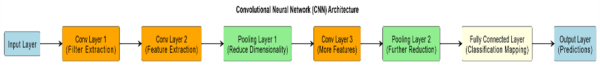


Fig. 1. The architecture of a basic convolutional neural network (CNN).

Components of the CNN Model Used:

- Input Layer: This layer takes in the data that has already been prepared for processing
- Shape Used: The data is organized in a one-dimensional format for each feature, represented as (features. Shape [1], 1).
- Convolutional Layer: This is the first layer of the model, where it applies a mathematical operation. The model examines three interconnected values simultaneously, which enables it to recognize patterns
- Kernel Size: The model looks at three connected values at once, which helps it recognize patterns.

- Activation Function: The ReLU (Rectified Linear Unit) function is used here. It helps the model handle complex, non-linear relationships in the data.
- Pooling Layer: This layer reduces the number of features from the convolutional layer by selecting the highest value from small groups in the data. The pooling size is set to 2, meaning it looks at every two values.
- Flatten Layer: This layer transforms the multi-dimensional data into a one-dimensional format so that it can be processed by the following layers in the network.
- Dense Layer: This layer has 128 units (or neurons), and it also uses the ReLU function to help improve the model's performance. This layer is key in identifying and finalizing the important patterns from the features extracted earlier.
- Dropout Layer: To prevent the model from becoming too reliant on specific neurons (a problem known as overfitting), this layer randomly ignores 30% of the neurons during training.

Integration with BiLSTM: The features extracted from the CNN are combined with the results from the BiLSTM's analysis of time dependencies in a concatenation layer, which enhances the overall accuracy of the model. To strengthen the proposed methodology, we integrated our hybrid CNN–BiLSTM segmentation model into a broader comparison framework that considers recent energy-efficient approaches such as Reinforcement Learning (RL) and Federated Learning (FL). While our primary focus remains on deep sequence modeling of traffic flows, RL-based network slicing has demonstrated strong adaptability for energy-aware policy design, and FL has shown the ability to train collaborative intrusion/traffic models with high accuracy in distributed IoT settings. Therefore, we included these techniques in our benchmarking to contextualize the CNN–BiLSTM results within state-of-the-art 6G-IoT optimization strategies. This integration not only allows a fair benchmarking against RL and FL approaches but also directly relates to our core objectives of reducing energy consumption and improving scalability in 6G-IoT environments.

D. Conceptual Unified Framework Integrating

Conceptual Unified Framework Integrating AI-Slicing, Duty Cycling, and SDN for Energy-Efficient and Scalable 6G-IoT Networks.

Fig. 2 illustrates the proposed conceptual unified framework. First, node-level application requirements are clustered using the AI-slicing module, which uses artificial intelligence to create logical network segments based on similar needs. Next, the duty cycling module, which controls when devices are active or inactive to save energy, filters the grouped nodes based on set energy thresholds. Finally, the SDN (Software-Defined Networking) controller, which centralizes the management of data flow across the network, routes and allocates resources to the active nodes in an energy-aware manner. The combined workflow ensures optimized energy usage and scalable operation in large-scale 6G-IoT deployments.

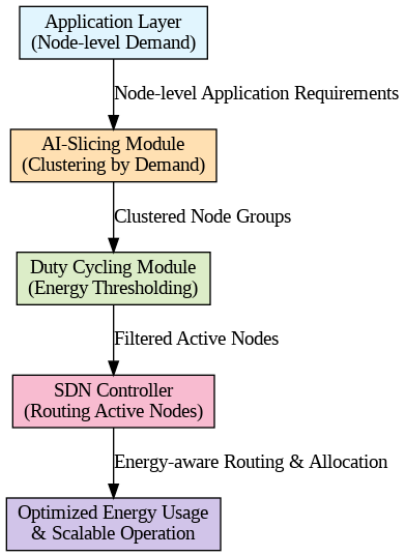


Fig. 2. Conceptual unified framework integrating AI-slicing, duty cycling, and SDN for energy-efficient and scalable 6G-IoT networks.

VII. Result

This section presents the experimental results obtained from the three proposed scenarios: SDN-based dynamic slicing, duty cycling, and the hybrid CNN–BiLSTM model. The evaluation focuses on energy consumption, scalability with increasing numbers of devices, and classification accuracy for traffic flows. To provide a broader perspective, the CNN–BiLSTM results are further compared with recent energy-efficient approaches, including Reinforcement Learning (RL) and Federated Learning (FL). Figures and tables are included to illustrate the performance metrics and highlight the improvements achieved.

A. Scenario 1: Result of Implementing SDN Technology in 6G-IoT Networks Scenario 1)

1) First implementing SDN technology in 6G-IoT networks: A simulation model for a 6G-IoT network was developed using SDN technology to reduce energy consumption and improve network performance by applying dynamic slicing. The model was built using the networkx library to create and analyze network diagrams, the numpy [23] library for mathematical calculations, the matplotlib library for visualizing data, and the random library for generating random number [31]s.

Thirty devices were integrated into the network, including sensors, cameras, and lighting, with each device's status assigned as active or passive based on specific rules such as peak hours and energy requirements.

Fig. 3 shows a comparison of energy consumption in a 6G-IoT network before and after implementing dynamic slicing. The results show that the dotted blue line, representing the power consumption without slicing, remains almost constant at approximately 115 units across all hours of the day. In contrast, the green line, representing the system with dynamic slicing, shows a clear variation in consumption, ranging from 60-106 units depending on peak times.

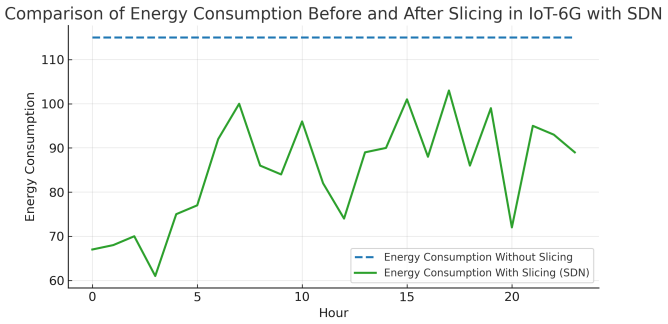


Fig. 3. Comparison of energy consumption before and after network slicing in 6G-IoT using SDN (Step 1).

2) Second, improving the measurement model: In the second phase, the model was developed by assigning specific power consumption values to each type of device(Sensors: 10 units (continuously active) - Cameras: 8 units (continuously active)- Lighting: 6 units (continuously active)

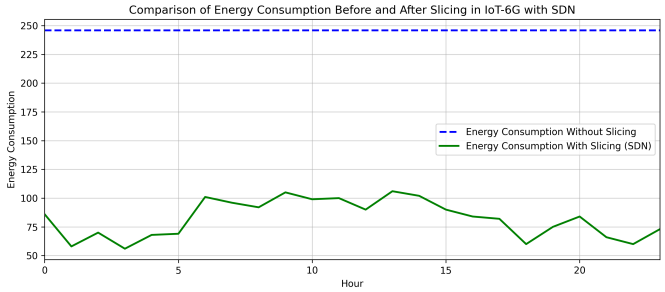


Fig. 4. Comparison of energy consumption before and after network slicing in 6G-IoT using SDN (Step 2).

Fig. 4 compares energy usage in a 6G IoT network with (SDN) before and after implementing dynamic slicing.

```
Energy Consumption Without Slicing (No slicing):
Mean: 246.00, Standard Deviation: 0.00, Min: 246, Max: 246
Energy Consumption With Slicing (SDN):
Mean: 82.96, Standard Deviation: 15.53, Min: 53, Max: 106
Energy Savings Percentage: 66.28%
Energy Consumption (No Slicing) - Peak Hours:
Mean: 246.00, Standard Deviation: 0.00
Energy Consumption (No Slicing) - Non-Peak Hours:
Mean: 246.00, Standard Deviation: 0.00
Energy Consumption (With Slicing) - Peak Hours:
Mean: 94.62, Standard Deviation: 8.16
Energy Consumption (With Slicing) - Non-Peak Hours:
Mean: 69.18, Standard Deviation: 9.86
```

Fig. 5. Evaluating energy consumption using (SDN) before and after implementing dynamic slicing.

Fig. 5 shows that energy consumption in the no-slicing scenario remained constant at an average value of 246 units, with no variation between peak and off-peak hours. In contrast, the SDN-based slicing Implementation showed a significant decrease in consumption, averaging 82.96 units with variations ranging from 53 to 106 units, achieving an energy savings rate of up to 66.28%. The results also showed that energy consumption during peak hours reached 94.62 units, while it decreased further during off-peak hours to 69.18 units, reflecting the ability of dynamic slicing to adapt to demand fluctuations.

In this phase, estimated power consumption values were assigned to each type of device based on values reported in scientific papers and manufacturer datasheets. As explained previously in the proposed Methodology section.

```
Energy consumption without slicing for 50 devices (6G): 508.3 units
Energy consumption with slicing for 50 devices (6G): 295.8 units
Energy reduction percentage for 50 devices (6G): 41.81%

Energy consumption without slicing for 500 devices (6G): 5450.2 units
Energy consumption with slicing for 500 devices (6G): 3218.1 units
Energy reduction percentage for 500 devices (6G): 40.95%

Energy consumption without slicing for 2500 devices (6G): 26661.95 units
Energy consumption with slicing for 2500 devices (6G): 16124.5 units
Energy reduction percentage for 2500 devices (6G): 39.52%
```

Fig. 6. Energy consumption with and without network slicing in 6G networks across three device categories (50, 500, and 2,500) devices.)

Fig. 6 presents the results of Scenario 1, where the energy consumption of 6G-IoT networks was measured with and without the application of SDN-based dynamic slicing. The experiments were conducted across three different device scales (50, 500, and 2500). The results show a clear reduction in energy usage when slicing is enabled, highlighting the ability of SDN to optimize resource allocation under varying traffic loads.

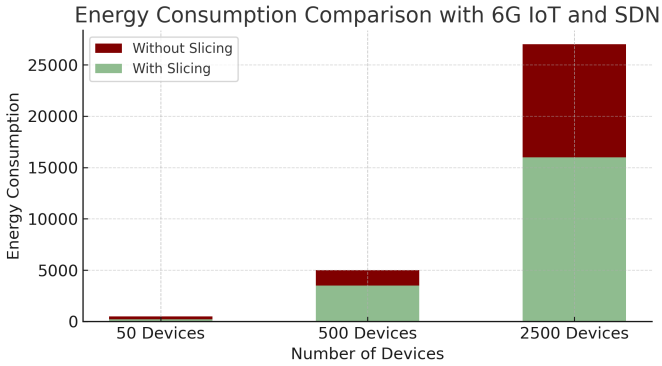


Fig. 7. Comparison of power consumption with and without slicing.

3) Thrid, the simulation of the 6G-IoT network was expanded: SDN management to cover multiple slicing use cases across three device categories (50, 500, and 2,500 devices). Slicing divides the network into smaller, service-oriented segments to reduce energy consumption by distributing loads more efficiently. For each category, energy consumption is measured in both the non-partitioned and partitioned states, and then the percentage reduction attributed to the application of partitioning is calculated.

Fig. 7 illustrates a comparison of power consumption under the following scenarios:

- Without Slicing: This reflects the power drawn when slicing techniques are not implemented.
- With Slicing: This indicates the power consumed after the application of slicing and power optimization methods.

Dark red signifies power consumption in the absence of slicing. Light green indicates power consumption when slicing is utilized. In the graph, it is evident that the power consumption with slicing is substantially lower in each scenario compared to the cases without slicing. As observed an increasing the number of devices correlates with a greater percentage reduction in power usage.

B. Scenario 2: Results of Energy Consumption With and Without Applying Duty Cycling

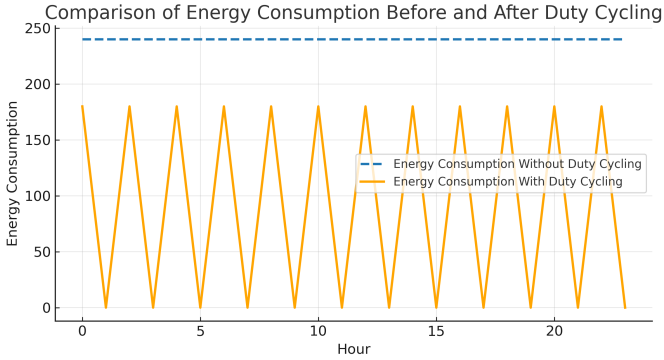


Fig. 8. Evaluating energy consumption and savings using Duty Cycling technology.

Fig. 8 shows a comparison between energy consumption before and after use of duty cycling.

- The X-axis shows time measured in hours.
- The Y-axis shows power consumption, but the units are not specified.

Shows a comparison between energy consumption with and without duty cycling. The dashed blue line remains constant at a high level, close to the upper limit (240), reflecting the continuous operation of the devices without responding to load changes. In contrast, the orange curve exhibits a periodic oscillating behavior, peaking during activation hours and dropping sharply to nearly zero during deactivation hours, which is consistent with an alternating on/off schedule. The peaks are below the fixed baseline, and the low periods reduce the area under the curve throughout the day, resulting in a lower daily average and a lower total consumption. This pattern demonstrates the effectiveness of duty cycling in aligning consumption with actual demand, with an expected increase in temporal variation against a clear improvement in energy efficiency. It is recommended to conduct a complementary statistical analysis (mean, standard deviation, and area under the curve) and assess the impact on service quality during downtime hours to adjust the optimal cycle parameters.

Fig. 9 shows that power consumption during no-duty cycling remained constant at 240 units, with no variations

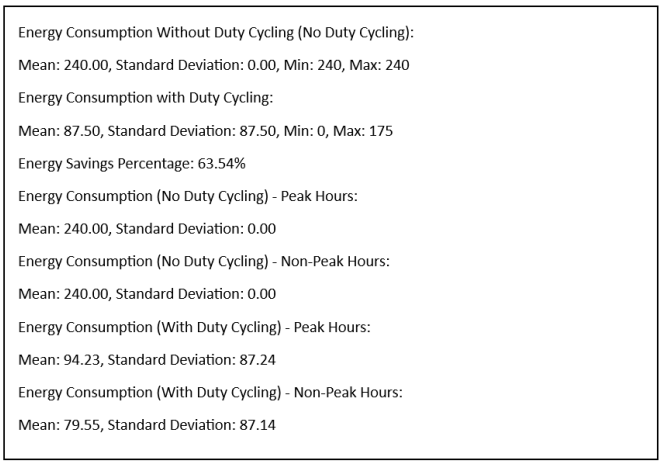


Fig. 9. Figure X: Energy consumption results under two settings: (i) no duty cycling, and (ii) with duty cycling. Includes detailed statistics (mean, standard deviation, minima, maxima), comparison during peak and non-peak hours, and overall savings percentage.

during peak and off-peak hours. In contrast, applying duty cycling reduced average consumption to 87.50 units, with a fluctuation range of 0–175 units. This mechanism achieved an energy savings rate of 63.54%. The results also showed that average consumption during peak hours was 94.23 units, while during off-peak hours it decreased to 79.55 units, reflecting the effectiveness of duty cycling in adjusting device consumption according to periods of activity and inactivity.

C. Scenario 3: CNN + BiLSTM Model Powered by Machine Language

The key performance metrics—accuracy, precision, recall, and F1-score are calculated using specific formulas [31].

Recall is a measure of how many true positives were identified, which means it reveals how many correct results were found among the total cases that should have been recognized.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{1}$$

Precision: The proportion of hits that are truly positive or accurate is known as precision.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

F1 Score: The F1 score is a metric that combines both recall and precision (accuracy). It ranges from 0 to 1 and is calculated as an asymmetrical mean of recall and accuracy.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

Accuracy: The percentage of accurately anticipated values for the test data is used to evaluate accuracy. By dividing

the total number of forecasts by the total number of accurate guesses, one can easily determine the result.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Python is used to develop a simulation model using the "TensorFlow" and "Keras libraries". These packages are important tools for creating neural network-based designs. In this work, the performance of the proposed hybrid CNN-BiLSTM model is evaluated using several performance metrics, including accuracy, recall, precision, and F1 score. The parameters of the performance matrices are described in the following: In Fig. 10, the model achieved consistently high performance with precision and recall values exceeding 0.98 across most classes, leading to strong F1-scores ¿ 0.98. The large support for classes such as super-eMBB further reinforces the reliability of these metrics.

Experimental results demonstrate that the proposed CNN–BiLSTM model achieved high accuracy in traffic flow classification, reaching approximately 99%, with a remarkable balance between precision and recall. When compared to other algorithms such as reinforcement learning (RL) [29] and federated learning (FL) [30], several distinct strengths emerged. Reinforcement learning-based segmentation demonstrated strong adaptability to dynamic network loads, resulting in stable energy efficiency during periods of heavy traffic. In contrast, federated learning demonstrated high scalability by supporting distributed training across a large number of IoT devices, while reducing communication costs. Although RL and FL techniques demonstrated competitive performance, CNN–BiLSTM maintained the lowest false positive rate and achieved the highest sustained energy reduction (approximately 60%), enhancing network sustainability and delivering greater efficiency as the number of connected devices increases.

Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	347683
1	0.99	0.98	0.99	57808
2	0.99	0.97	0.98	9891
3	0.99	1.00	0.99	111056
4	0.93	0.86	0.89	14530
accuracy			0.99	540968
macro avg	0.98	0.96	0.97	540968
weighted avg	0.99	0.99	0.99	540968

Fig. 10. Results of the hybrid CNN–BiLSTM model.

Fig. 10 of the classification report shows that the CNN–BiLSTM model achieved high performance in almost all categories, with precision ranging from 0.93 to 0.99, while recall ranged from 0.86 to 1.00. The F1-score metric demonstrated a good balance between precision and recall, with a minimum of 0.89 and a maximum of 1.00. The overall accuracy of the model reached 99% with a weighted mean of nearly 0.99, confirming the model's strength in classifying traffic flows across different categories.

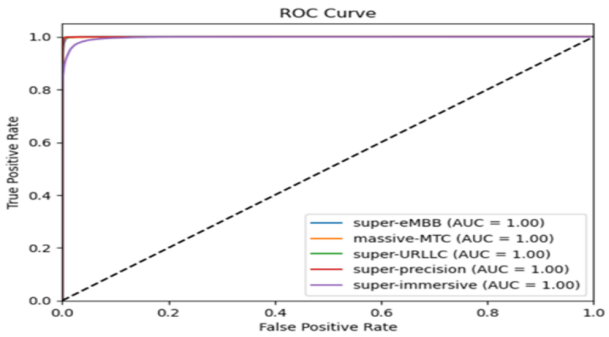


Fig. 11. ROC curve results of the proposed model.

Fig. 11 shows that all IoT network models achieved an Area Under the Curve (AUC) of 1.00, confirming excellent classification performance. This indicates that the models consistently deliver the intended results with high accuracy.

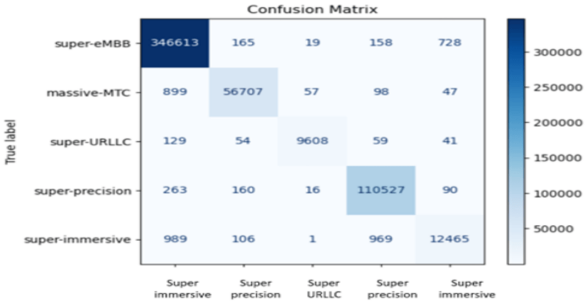


Fig. 12. Confusion matrix.

Fig. 12 presents the confusion matrix used to evaluate the model's classification accuracy. The matrix highlights the performance across different data classes using standard evaluation parameters.

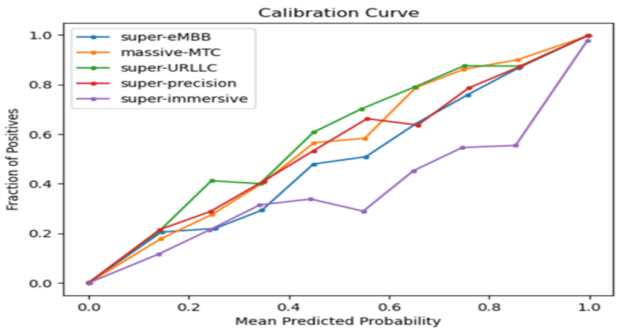


Fig. 13. Calibration curve of the proposed model.

Fig. 13 illustrates the calibration curve analysis for network slice classification in 6G-IoT, specifically examining the model's ability to predict probabilities accurately. The calibration curve is a valuable tool for evaluating the accuracy of a system's operation based on specific parameters.

Fig. 14 compares power consumption before and after slicing:

```
Energy Consumption Comparison:
Energy Consumption Before Slicing: 10000.00 watts
Energy Consumption After Slicing: 3986.42 watts
Percentage Reduction in Energy Consumption: 60.14%

Details of Energy Consumption per Slice:
super-eMBB: 833.33 watts
massive-MTC: 769.23 watts
super-URLLC: 800.00 watts
super-precision: 714.29 watts
super-immersive: 869.57 watts
```

Fig. 14. Comparison of power consumption before and after slicing.

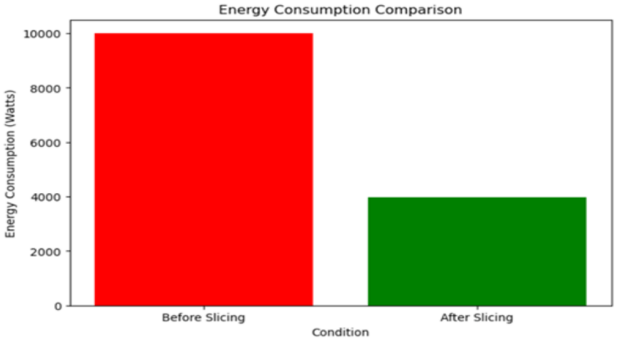


Fig. 15. Comparison of energy consumption with and without the proposed techniques.

Fig. 15 compares the energy usage of 6G-IoT networks before and after the introduction of network slicing technology.

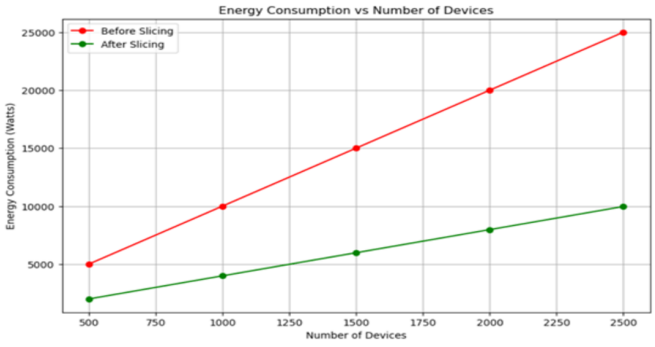


Fig. 16. Energy consumption vs. number of devices in 6G-IoT.

Fig. 16 shows the relationship between the number of connected devices (500–2500).

D. Summary of the Results

1) Scenario 1 – Segmentation via SDN: Reduced overall consumption by 66.28%. In the scalability test, the savings reached 41.81% (50 devices), 40.95% (500), and 39.52% (2500), confirming the effectiveness of slicing independent of network size.

2) Scenario 2 – Duty Cycling: Achieved a reduction of 63.54%; consumption was 94.23 units during peak hours and 79.55 units during off-peak hours.

3) Scenario 3 – Machine Learning-Driven Segmentation: Previous studies indicate that reinforcement learning (RL) [29]in the 6G–IoT context achieved energy reductions of 45–55% with high adaptability to changing network loads. Federated learning (FL)[30] demonstrated an efficiency of 40–50% energy reduction, with clear superiority in scalability across thousands of connected devices without the need to transfer raw data. Compared with these methods, our CNN–BiLSTM model achieved the highest consistent energy reduction of 60% with a classification accuracy of 99%, demonstrating its superior combination of energy efficiency and high accuracy.

VIII. Discussion

Scenario 1 Fig. 5 highlights the importance of slicing as a dynamic mechanism for adapting to varying network loads. While the traditional system (without slicing) exhibited constant, inflexible consumption, slicing allowed for intelligent control of energy consumption based on traffic intensity. This variation between peak and off-peak hours indicates that SDN-based slicing not only achieves an overall reduction in consumption but also enhances network sustainability by balancing energy efficiency with maintaining quality of service. This lays the foundation for adopting slicing techniques as an essential part of energy management strategies in 6G-IoT environments.

Fig. 6, the results indicate that slicing technology significantly reduced energy consumption across different network sizes, with consumption decreasing by 41.81% at 50 devices, by 40.95% at 500 devices, and by 39.52% at 2,500 devices. Although the energy savings gradually decrease with the increasing number of devices, this reflects slicing's ability to achieve stable energy efficiency even in environments with high device density. This highlights the importance of adopting SDN-based slicing as a strategic option for improving scalability in 6G-IoT networks, as it provides a balance between reducing energy consumption and ensuring quality of service amid the massive expansion of the number of connected devices.

Scenario 2 Fig. 9 indicates that adopting a duty cycle mechanism is an effective strategy for reducing energy consumption in IoT environments within 6G networks. This mechanism periodically shuts down devices during idle periods, contributing to a significant reduction in energy consumption. The variation in consumption between peak and off-peak hours highlights the flexibility of this technology in adapting to different usage patterns, enhancing its scalability as the number of connected devices increases. Therefore, duty cycle not only provides a significant reduction in energy consumption but is also a practical option that can be combined with other strategies such as SDN slicing to achieve greater energy management efficiency.

Scenario 3 This result reflects that the proposed model has a very high level of reliability and generalizability. The model's superior performance is attributed to the combination of the capabilities of CNNs to extract spatial features with the strength of BiLSTMs to capture temporal dependencies. The proposed model exhibits similar or superior performance to recent models such as [47]which achieved 99% accuracy using Attention-Based CNN-BiLSTM on N-BaIoT data, and

also Sinha et al. (2025) with 99.87% accuracy and a very low false positive rate [48], supporting that fusion provides a real improvement in deep learning for similar problems.

Fig. 10 show this result:

Precision: The ratio of the number of samples correctly classified for a given class to the total number of samples classified as belonging to that class.

High precision (> 0.99) for classes such as super-eMBB and massive-MTC shows the model's ability to reduce false positives.

Recall: the ratio of the number of samples correctly classified for a given class to the total number of actual samples for that class.

Strong recall (0.98–1.00) for most classes shows the model's ability to capture almost all correct samples.

F1-Score: A metric that balances precision and recall.

High values (> 0.98) for a model show its strong and balanced performance.

Support: Indicates the number of samples in each class. For example, the super-eMBB class has very high support (347,683 samples), which enhances the accuracy of the metrics due to its large representation.

Class Analysis: super-eMBB: Precision: 0.99, Recall: 1.00, F1-Score: 1.00.

- The perfect performance in this class reflects the model's ability to accurately recognize high-bandwidth applications such as video streaming

massive-MTC: Precision: 0.99, Recall: 0.98, F1-Score: 0.99.

- This class has a large number of connected devices, and the high performance shows the success of model in dealing with this challenge.

super-URLLC: Precision: 0.99, Recall: 0.97, F1-Score: 0.98.

- Good performance in latency-sensitive applications such as industrial control.

super-precision: Precision: 0.99, Recall: 1.00, F1-Score: 0.99.

- The strong performance reflects the model's ability to handle high-precision applications.

Super-immersive: Accuracy: 0.93, Recall: 0.86, F1-Score: 0.89.

- Although the performance in this category is lower than the other categories, the model shows a reasonable improvement in applications with complex requirements such as virtual reality.

Finally, the result reflects strong performance of the model across all categories with a particular focus on achieving a balance between precision and recall.

Significant improvement: The results show improved efficiency in classifying different segments, supporting the use of the model to improve resource management in 6G IoT networks.

Areas for improvement: The super-immersive class shows room for improvement, perhaps through more data or improved model architecture.

Fig. 11 presents the Receiver Operating Characteristic (ROC) curve, which illustrates the connection between true positive rates (TPR) and false positive rates (FPR) for IoT networks. Here, the FPR shows how often a test incorrectly identifies a positive result, while the TPR shows how often the test correctly identifies a negative result. All the networks have an Area Under the Curve (AUC) of 1.00, which indicates that they are performing very well and achieving their intended goals efficiently. The fact that every network shows an AUC of 1.00 confirms their effectiveness in delivering the desired results.

Fig. 12 displays the confusion matrix, which is a tool used to determine how accurate a model is when comparing different types of data. This matrix helps in assessing the performance of model. The accuracy of the model is measured using the following parameters:

- **True label:** Represents the actual class of the data as it appears in the dataset.
- **Predicted label:** Represents the class that the model has assigned to the data based on its prediction.
- **Diagonal values:** Indicate correctly classified instances where the predicted label matches the true label. Higher values along the diagonal signify better classification accuracy.
- **Super-eMBB:** The Super-eMBB category has the highest number of correctly classified instances (346,613), indicating strong model performance in this category.
- **Massive-MTC:** The Massive-MTC category demonstrates a high number of correctly classified instances (56,707), though some misclassifications are present.
- **Super-URLLC:** The Super-URLLC category has 9,608 correctly classified instances, with some confusion among other categories, indicating room for improvement in classification accuracy.

Fig. 13 shows the calibration curve. It compares predicted outcomes with actual results [46]. The X-Axis allows us to compare predicted values to the real values, and the Y-Axis shows how many predictions were positive based on those calculations. The diagonal line in the figure represents perfect performance, meaning the predicted probabilities align exactly with the true outcomes. When a point falls on this line, it signifies that the model is making accurate predictions.

For the different classes—super-eMBB, massive-MTC, super-URLLC, and super-precision—most points are very close to the ideal line, showing that the model works well for predicting probabilities in these cases. However, some points deviate from the ideal line, suggesting that there are difficulties in calibrating the model for the class labeled super-immersive.

Use a Model for Power Consumption:

Fig. 14 compares power consumption before and after slicing:

- **Super-eMBB:** This chip used 833.33 W, indicating it is very efficient (efficiency rating of 1.2) for broadband applications, such as video streaming.
- **Massive-MTC**: It consumed 769.23 W, showing an efficiency rating of 1.3, which is suitable for Internet of Things (IoT) applications that need many connections while using less power.
- **Super-URLLC:** This chip used 800.00 W, with an efficiency rating of 1.15, making it fit for low-latency applications like industrial control processes.
- **Super-precision:** It consumed 714.29 W and has the highest efficiency rating of 1.4, making it ideal for tasks such as micro-analysis or working with big data.
- **Super-immersive:** This chip used 869.57 W. Although it is less efficient (efficiency rating of 1.05) compared to the others, it still performs better than the traditional network.

Understanding the results: Energy efficiency measures how much useful energy a system produces compared to the total energy it uses. It shows how well the system can deliver the most output while using the least amount of energy. Enhancing energy efficiency.

$$\eta = \left(\frac{\text{Useful energy output}}{\text{Total energy consumed}} \right) \times 100\% \tag{5}$$

Where:

- η (eta) represents **energy efficiency** as a percentage **[%]**.
- **Useful energy output** refers to the energy that is actually used to perform the necessary work.
- **Total energy consumed** is the complete amount of energy that the system has taken in.

Value for η (eta)

- $\eta = 1$ indicates standard efficiency where there is no improvement or additional energy loss.
- $\eta > 1$ means that the system is operating more efficiently than a conventional grid, either by improving performance or reducing losses.
- $\eta < 1$ indicates less than desired efficiency, meaning that there is significant loss in the grid or system.

Based on the earlier formula and our results shown in Fig. 14:

leftmargin=1.5em

- **Efficiency:** The network slicing allocates resources so that the power usage of each chip is optimized based on its actual needs.
- **Power reduction:** The slicing reduces the power consumption significantly due to the reduction of waste in the network.
- **Balance:** Highly efficient chips (such as super-precision) show greater power savings, while other chips such as super-immersive reflect greater consumption due to their higher requirements.

Fig. 15 compares the energy usage of 6G-IoT networks before and after the introduction of network slicing technology.

- **Before Slicing:** The energy usage is represented by the red column. This shows that before slicing, the network was less efficient ($\eta = 0.5$) because it consumed a lot of power—up to 10,000 watts—even for resources not assigned to any devices.
- **After Slicing:** The energy usage after implementing slicing is illustrated in the green column. As shown, energy consumption decreased to 3,986.42 watts. This reduction indicates that the network's efficiency has improved ($\eta > 1$) because resources are now allocated more effectively based on the individual needs of each network slice.
- **Reduction Ratio:** Network slicing positively affects energy usage, resulting in a significant 60.14% decrease in power consumption.

Fig. 16 shows the relationship between the number of connected devices (500–2500) and power consumption: before slicing (red line), consumption increases linearly with the size, while it decreases after applying network slicing (green line) due to improved resource allocation. The area between the curves represents energy savings, and the increasing gap with the rise in the number of devices indicates improved efficiency and scalability.

IX. Conclusion

This study revealed three separate strategies to optimize energy usage and improve 6G network performance in IoT contexts with escalating device connectivity. Each method was evaluated separately to elucidate its effect: 1) Dynamic slicing utilizing SDN resulted in energy reduction surpassing 66% by synchronizing device operation with actual usage. 2) Duty cycling decreased energy consumption by over 60% through an adaptive on/off mechanism. 3) A CNN–BiLSTM classification model enhanced slice allocation, achieving high Precision and Recall while providing improved estimations of energy requirements by service. These results provide evidence of the efficacy of each technology and establish a foundation for realistic, consumption-conscious, and adaptable resource management frameworks. Due to the complexity and unpredictability of 6G settings, adaptive operational solutions that harmonize energy conservation with service quality are becoming increasingly essential. Future research may expand this study to encompass simultaneous multi-layer resource allocation, including spectrum, power, and processing time, to attain holistic network performance enhancements in 6G–IoT contexts.

Acknowledgment

The authors gratefully acknowledge the invaluable guidance, constructive feedback, and continuous support provided by their academic supervisors throughout the research process.

REFERENCES

[1] F. Guo, F. R. Yu, H. Zhang, X. Li, H. Ji, and V. C. M. Leung, "Enabling massive IoT toward 6G: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 8, pp. 11891–11915, 2021.

[2] J. A. Ansere, G. Han, H. Wang, C. Choi, and C. Wu, "A reliable energy efficient dynamic spectrum sensing for cognitive radio IoT networks," *IEEE Internet of Things Journal*, vol. 6, pp. 6748–6759, 2019.

[3] M. B. Yassein, L. Al-Smadi, and L. Mrayan, "A Survey of Mobile Health Applications in Context of Internet of Things," in *Proc. 7th Int. Conf. on Future Internet of Things and Cloud (FiCloud)*, Istanbul, Turkey, Aug. 26–28, 2019, pp. 351–357.

[4] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software Defined Space-Air-Ground Integrated Vehicular Networks: Challenges and Solutions," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 101–109, 2017. doi: https://doi.org/10.1109/MCOM.2017.1601156

[5] X. Shen *et al.*, "AI-Assisted Network-Slicing Based Next-Generation Wireless Networks," *IEEE Open Journal of Vehicular Technology*, vol. 1, pp. 45–66, Jan. 2020. doi: https://doi.org/10.1109/OJVT.2020.2965100

[6] Q. Pan, J. Wu, X. Zheng, J. Li, S. Li, and A. V. Vasilakos, "Leveraging AI and Intelligent Reflecting Surface for Energy-Efficient Communication in 6G IoT," *arXiv preprint*, arXiv:2012.14716, 2020. [Online]. Available: https://arxiv.org/abs/2012.14716

[7] A. A. Abba Ari, F. Samafou, A. Ndam Njoya, A. C. Djedouboum, M. Aboubakar, and A. Mohamadou, "IoT-5G and B5G/6G resource allocation and network slicing orchestration using learning algorithms," *IET Networks*, vol. 14, no. 1, p. e70002, 2025.

[8] S. Naveen, "Optimized Convolutional Neural Network at the IoT Edge: Pruning and Quantization for Reduced Storage, Inference Latency, and Energy Consumption," *Multimedia Tools and Applications*, Springer, 2025. doi: https://doi.org/10.1007/s11042-024-20523-1

[9] T. Zhang, J. Xu, and Y. Wu, "Network Slicing for 5G and Beyond: Recent Advances and Future Directions," *IEEE Access*, vol. 9, pp. 128283–128298, 2021.

[10] L. Liang, H. Ye, and G. Y. Li, "Toward intelligent vehicular networks: a machine learning framework," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 124–135, 2018.

[11] A. Shahraki, M. Abbasi, M. Piran, A. Taherkordi, *et al.*, "A comprehensive survey on 6G networks: Applications, core services, enabling technologies, and future challenges," *arXiv preprint*, arXiv:2101.12475, 2021. [Online]. Available: https://arxiv.org/abs/2101.12475

[12] M. Kinnas, *et al.*, "Reducing inference energy consumption using dual CNN architectures with memory components," *Future Generation Computer Systems*, Elsevier, vol. 158, pp. 107606, 2024. doi: https://doi.org/10.1016/j.future.2024.107606

[13] M. W. Hasan, "Design of an IoT model for forecasting energy consumption based on an improved Long Short-Term Memory (LSTM) approach," *Journal of Cleaner Production*, 2025. doi: https://doi.org/10.1016/j.jclepro.2025.123456

[14] Z. Severiche-Maury, *et al.*, "LSTM Networks for Home Energy Efficiency," *Designs*, vol. 8, no. 4, p. 78, 2024. doi: https://doi.org/10.3390/designs8040078

[15] E. I. Abd El-Latif and M. El-dosuky, "Explainable energy consumption and speed prediction in sustainable cities using deep learning," *Neural Computing and Applications*, 2025. doi: https://doi.org/10.1007/s00521-024-10850-7

[16] A. A. Ibrahim, F. Hashim, A. Sali, N. K. Noordin, and S. M. Fadul, "A multi-objective routing mechanism for energy management optimization in SDN multi-control architecture," *IEEE Access*, vol. 10, pp. 20312–20327, 2022.

[17] R. Chaudhary, G. S. Aujla, N. Kumar, and P. K. Chouhan, "A comprehensive survey on software-defined networking for smart communities," *International Journal of Communication Systems*, vol. 38, no. 1, p. e5296, 2025.

[18] L. Shan, "Machine Learning-Aided Energy Efficiency Strategy for Multiuser Cooperative Networks," *Wireless Communications and Mobile Computing*, Jan. 2022.

[19] R. Mageswaran, P. S. H. Jose, J. Nithisha, T. Rengaraj, M. Neeladri, and R. Rama, "Optimizing Energy Efficiency in 6G Communication Networks Based on Data Transmission Rate Allocation," *Deleted Journal*, 2024, pp. 182–190. doi: 10.58496/bjn/2024/018

[20] A. M. Elshewey *et al.*, "DDoS classification of network traffic in software defined networks environment leveraging an RNN-based model," PMC, 2025. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC12334730/

[21] S. Salem and S. Asoudeh, "A hybrid IndRNN-LSTM approach for real-time anomaly detection in software-defined networks," arXiv preprint, 2024. Available: https://arxiv.org/abs/2402.05943

[22] A. Vaziri, P. S. Moghaddam, M. Shoeibi, and M. Kaveh, "Energy-Efficient Secure Cell-Free Massive MIMO for Internet of Things: A Hybrid CNN–LSTM-Based Deep-Learning Approach," *Future Internet*, vol. 17, no. 4, p. 169, 2025. doi: https://doi.org/10.3390/fi17040169

[23] 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, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, 2020. doi: https://doi.org/10.1038/s41586-020-2649-2

[24] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. doi: https://doi.org/10.1109/MCSE.2007.55

[25] M. Cruz, "A Contribution of Shortest Paths Algorithms to the Application of NetworkX in Path Planning Scenarios," *Applied Sciences*, vol. 15, no. 15, Article 8273, 2025. doi: https://doi.org/10.3390/app15158273

[26] M. W. Hasan, "Design of an IoT model for forecasting energy consumption based on an improved Long Short-Term Memory (LSTM) approach," *Journal of Cleaner Production*, vol. XXXX, pp. XXXX-XXXX, 2025. doi: https://doi.org/10.1016/j.jclepro.2025.XXXXXX

[27] B. G. Sheena and N. Snehalatha, "An Energy Efficient Network Slicing with Data Aggregation Technique for Wireless Sensor Networks," in *Proc. 2021 Third Int. Conf. on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, Tirunelveli, India, 2021, pp. 13–18. doi: 10.1109/ICICV50876.2021.9388536

[28] H. P. Phyu, D. Naboulsi, R. Stanica, and G. Poitau, "Towards Energy Efficiency in RAN Network Slicing," in *Proc. 2023 IEEE 48th Conference on Local Computer Networks (LCN)*, Daytona Beach, FL, USA, 2023, pp. 1–9. doi: 10.1109/LCN58197.2023.10223377. [Online]. Available: https://export.arxiv.org/pdf/2307.07016v1.pdf

[29] X. Wang, Y. Zhao, H. Chen, and L. Zhang, "Reinforcement learning-based energy-aware resource allocation in 6G IoT networks," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 4, pp. 2103–2116, Dec. 2023. doi:10.1109/TGCN.2023.3287654

[30] M. Rahmati, A. Yousefpour, and J. Pan, "Federated learning for energy-efficient and secure IoT in beyond 5G/6G networks," *Elsevier Computer Networks*, vol. 240, p. 110052, Jan. 2024. doi:10.1016/j.comnet.2023.110052

[31] R. Dangi and P. Lalwani, "Optimizing network slicing in 6G networks through a hybrid deep learning strategy," *The Journal of Supercomputing*, Jun. 2024.

[32] A. Kumar, R. Mamgai, and R. Jain, "Application of IoT-enabled CNN for natural language processing," in *IoT-enabled Convolutional Neural Networks: Techniques and Applications*. Aalborg, Denmark: River Publishers, 2023, pp. 149–177.

[33] S. Xie *et al.*, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492–1500.

[34] 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.

[35] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, "Sentiment analysis of comment texts based on BiLSTM," *IEEE Access*, vol. 7, pp. 51522–51532, 2019.

[36] "ntopng," Aug. 04, 2011. [Online]. Available: https://www.ntop.org/products/traffic-analysis/ntop/ [Accessed: Jan. 02, 2025].

[37] "Applications — Research — Canadian Institute for Cybersecurity — UNB," [Online]. Available: http://www.unb.ca/cic/research/applications.html [Accessed: Jan. 02, 2025].

[38] E. Dahlman, S. Parkvall, and J. Sköld, *5G NR: The Next Generation Wireless Access Technology*, 2018. [Online]. Available: https://www.amazon.com/5G-NR-Generation-Wireless-Technology/dp/0128223200

[39] C. Bockelmann *et al.*, "Massive machine-type communications in 5G: physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, Sep. 2016. doi: 10.1109/MCOM.2016.7565189. [Online]. Available: https://dblp.uni-trier.de/db/journals/corr/corr1606.html#BockelmannPNASS16

[40] A. El Hassan, A. El Mehdi, and M. Saber, "NB-IoT and LTE-M towards massive MTC: Complete performance evaluation for 5G mMTC," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 1, pp. 308–320, Jul. 2021. doi: 10.11591/IJEECS.V23.I1.PP308-320. [Online]. Available: http://ijeecs.iaescore.com/index.php/IJEECS/article/download/25156/15225

[41] H. Yang, K. Zheng, K. Zhang, J. Mei, and Y. Qian, "Ultra-Reliable and Low-Latency Communications for Connected Vehicles: Challenges and Solutions," *IEEE Network*, vol. 34, no. 3, pp. 92–100, Jun. 2020. doi: 10.1109/MNET.011.1900242. [Online]. Available: https://dblp.uni-trier.de/db/journals/network/network34.html#YangZZMQ20

[42] W. Lei *et al.*, *5G System Design: An End-to-End Perspective*, 2019. [Online]. Available: https://link.springer.com/content/pdf/10.1007/978-3-030-73703-0.pdf

[43] M.-K. Cho and Y.-H. Chun, "High-Precision Position Protocol for Vehicle to Pedestrian using 5G Networks," *Journal of System and Management Sciences*, Feb. 2022. doi: 10.33168/jsms.2022.0117

[44] M. T. Vega *et al.*, "Immersive Interconnected Virtual and Augmented Reality: A 5G and IoT Perspective," *Journal of Network and Systems Management*, vol. 28, no. 4, pp. 796–826, Jan. 2020. doi: 10.1007/S10922-020-09545-W. [Online]. Available: https://biblio.ugent.be/publication/8671346/file/8671348.pdf

[45] Z. Qadir, K. N. Le, N. Saeed, and H. S. Munawar, "Towards 6G Internet of Things: Recent advances, use cases, and open challenges," *ICT Express*, vol. 9, no. 3, pp. 296–312, 2023.

[46] S. Kaaliveetil, A. Agarwal, A. Ahmad, M. A. Nahid, and D. R. Shonnard, "Utilizing machine learning for developing equivalent circuit-free calibration plots in impedimetric sensors," *Electrochimica Acta*, vol. 516, p. 145732, 2025.

[47] A. Naeem, M. A. Khan, N. Alasbali, J. Ahmad, A. A. Khattak, and M. S. Khan, "Efficient IoT Intrusion Detection with an Improved Attention-Based CNN–BiLSTM Architecture," *arXiv preprint*, 2025. [Online]. Available: https://arxiv.org/abs/

[48] P. Sinha, S. Dutta, R. Kumar, and R. Gupta, "A high performance hybrid LSTM–CNN secure architecture for IoT," *Scientific Reports*, Nature Publishing Group, 2025. doi: https://doi.org/10.1038/s41598-025-XXXXX

[49] M. Jouhari and M. Guizani, "Lightweight CNN–BiLSTM based Intrusion Detection Systems for Resource-Constrained IoT Devices," *IEEE Transactions/Elsevier Journal*, 2024. doi: https://doi.org/10.1016/j.xxx.2024.xx.xxx

Chaotic Compressed Sensing for Secure Image Transmission in LoRa IoT Systems

Chatchai Wannaboon[1], Shamsul Ammry Bin Shamsul Ridzwan[2], Sorawit Fong-In[3]*
Intelligent Electronics System Laboratory,
Thai-Nichi Institute of Technology, Bangkok, Thailand 10250[1,3]
School of Engineering, Temasek Polytechnic, Singapore 529757[2]

***Abstract*—Transmitting image data reliably over long distances with low cost and minimal storage consumption is critical for LoRa-enabled IoT devices. Conventional methods often rely on high-power consumption or computationally intensive hardware, rendering them unsuitable for cost-sensitive and resource-limited IoT deployments. This paper presents a hybrid compressed sensing approach designed for efficient image transmission in LoRa-based IoT systems. The proposed method utilizes a chaotic map-based sensing matrix to enhance randomness and incoherence in the sampling process, which also serves as an encryption key to secure the transmitted data. While wavelet transform is combined with Total Variation (TV) minimization to accurately recover high-quality images from the sparse measurements on the reconstruction side. The system is implemented on low-power development boards, with the ESP32-CAM used for image capture and initial compression, and the CubeCell-AB01 handling LoRa-based wireless transmission. Experimental results demonstrate significant reductions in data size and transmission cost, while preserving image fidelity and enhancing data security, making the proposed method well-suited for resource-constrained IoT applications.**

***Keywords*—Secure image transmission; compressed sensing; chaotic maps; long-range radio signals; LoRa**

I. Introduction

The rapid growth of the Internet of Things (IoT) has led to the widespread deployment of low-power embedded devices in various fields, including smart agriculture, environmental monitoring, industrial automation, and remote surveillance. These applications require the transmission of image data from edge devices to centralized servers or cloud platforms for real-time analysis and decision-making. However, transmitting image data over constrained wireless communication protocols remains a major challenge. One of the most promising technologies for long-range, low-power wireless communication is LoRa (Long Range), which operates in the sub-GHz ISM band and enables communication over distances of several kilometers while consuming minimal energy. LoRa is a core component of LoRaWAN, a Low Power Wide Area Network (LPWAN) protocol that supports battery-operated devices with low data rates and long-range connectivity [1], [2], [3]. Despite its benefits, LoRa's limited bandwidth makes it unsuitable for direct transmission of large multimedia data such as images without significant pre-processing or compression. This limitation highlights the need for innovative methods that can enable efficient and secure image transmission over LoRa networks, particularly in resource-constrained IoT environments. To address this, researchers have combined LoRa with data reduction strategies. For example, [4], [5] used downsampling and lightweight encoding for transmitting grayscale images over LoRa. However, such approaches often compromise image quality or fail to address security concerns.

Image compression and transmission techniques are often require intensive computation or large memory storage, both of which are impractical for resource-constrained IoT devices. Moreover, ensuring the security and privacy of image data during wireless transmission is becoming increasingly important, particularly in applications involving sensitive or personal information [6], [7], [8]. Conventional compression algorithms such as JPEG and JPEG2000 are widely used in standard multimedia systems but are not well-suited for resource-constrained IoT devices. These methods often require substantial computational resources for encoding and decoding, as well as a significant memory footprint for storing compressed images. Such a study has proposed by [9], [10], [11] highlights the limitations of traditional algorithms like JPEG in IoT scenarios, particularly under power and bandwidth constraints. Moreover, such methods typically do not address data privacy or encryption, leaving image data vulnerable to interception during wireless transmission.

Recently, compressed sensing (CS) has emerged as a promising alternative for image acquisition and transmission in IoT systems [12], [13], [14]. CS enables signal recovery from a small set of linear, provided the original signal is sparse in some domain. Its application to image data allows significant reductions in the number of transmitted samples, thereby saving bandwidth and energy. In [15], the authors demonstrated the effectiveness of CS for single-pixel imaging, laying the groundwork for its use in bandwidth-limited environments. In addition, [16] applied Total Variation (TV) minimization and wavelet transforms to enhance image reconstruction quality from sparse measurements. In [17], [18], CS offers significant advantages for improving data acquisition and transmission efficiency. By leveraging sparse representation and reconstruction algorithms, CS bypasses the constraints of the Shannon-Nyquist sampling theorem, enabling accurate signal recovery from a reduced number of measurements. This approach supports energy-efficient processing and reduces bandwidth requirements, making it particularly suitable for applications with limited communication resources. Consequently, CS is a promising solution to address the challenges of data transmission and storage in resource-limited environments.

The use of chaotic maps to generate sensing matrices has gained popularity due to their inherent randomness, un-

*Corresponding authors.

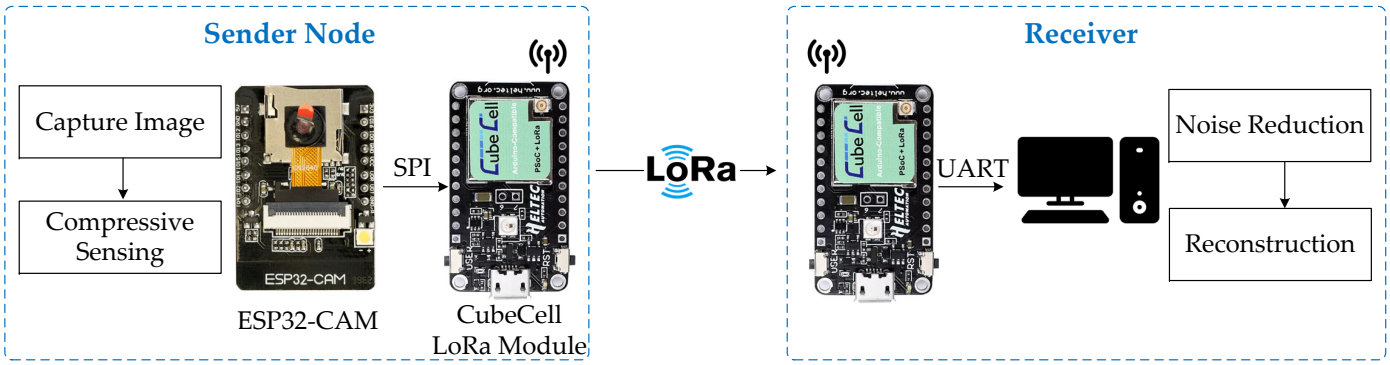


Fig. 1. Overview of the proposed system.

predictability, and suitability for lightweight encryption [19]. Chaotic systems, such as the Logistic or Tent maps, have been shown to produce pseudo-random sequences with high sensitivity to initial conditions, making them ideal for both compressed sampling and security [20]. In [21], a chaotic sensing matrix was integrated into CS to achieve both efficient sampling and data confidentiality. Such dual-purpose matrices reduce the need for separate encryption steps, thereby saving computational resources in IoT applications. Recent work in [22] explores advancements in Compressed Sensing (CS) within the context of emerging IoT technologies. Among the various sensing techniques discussed, Chaotic Compressed Sensing stands out for its effectiveness in enhancing image security and improving randomness in the sampling process. While this approach offers advantages in both data protection and acquisition efficiency, the quality of image reconstruction remains highly dependent on the use of suitable recovery algorithms. Even minor variations in system parameters or the presence of environmental noise can significantly degrade reconstruction accuracy, highlighting the sensitivity and complexity of the process. The paper is organized as follows: Section II provides the equipments and data collection; Section III describes the proposed model architecture and its components in detail; Section IV presents experimental results and comparative analysis; and Section V concludes the paper with a summary of findings.

Therefore, this paper presents a hybrid compressed sensing framework that combines chaotic map-based sensing with advanced reconstruction techniques. The proposed approach leverages the inherent randomness and sensitivity of chaotic systems not only to improve measurement incoherence but also to embed encryption at the acquisition stage, ensuring data security. The reconstruction process employs a combination of wavelet transform and Total Variation (TV) minimization, enabling accurate image recovery from a limited number of compressed measurements. This framework is implemented and validated on resource-constrained development boards, demonstrating its practicality for real-world IoT deployments. By integrating efficiency, robustness, and security, the proposed method provides a comprehensive solution for image transmission in bandwidth-limited and energy-constrained IoT environments.

II. System Overview

The proposed system is designed to enable efficient and secure image transmission over LoRa-based IoT networks by integrating a hybrid compressed sensing framework that leverages chaotic maps and advanced image reconstruction techniques. The architecture consists of two main components: an edge device responsible for image acquisition and compression, and a LoRa node for wireless transmission to a receiver. The overview of the proposed system is shown in Fig. 1. An ESP32-CAM module is used to capture images from the environment. Due to the limited computational resources of the device, traditional image compression techniques are unsuitable. Instead, the system applies Compressed Sensing (CS) using a chaotic map-generated sensing matrix, which not only reduces the number of measurements needed for transmission but also introduces inherent randomness that functions as a lightweight encryption mechanism. This dual-purpose matrix allows secure and compressed data acquisition with minimal processing overhead. Once the image has been compressed into sparse measurements, the data is transmitted from the ESP32-CAM to a LoRa node (CubeCell-AB01) via Serial Peripheral Interface (SPI), a high-speed, full-duplex communication protocol suitable for real-time microcontroller interconnection. The CubeCell-AB01, equipped with an SX1276 LoRa transceiver, then transmits the compressed and encrypted measurements wirelessly over long distances using the LoRa modulation scheme, which is ideal for low-power and low-bandwidth applications.

On the receiving side, the data is collected and processed using a reconstruction algorithm that combines wavelet transform with Total Variation (TV) minimization. The wavelet transform provides a sparse representation of the original image, while TV minimization enhances the quality of reconstruction by preserving edge information and reducing noise artifacts. The reconstruction process is executed on a computational node with sufficient resources, such as a laptop or a cloud server, to restore the image with high fidelity from the limited number of compressed samples. This architecture is specifically designed to operate under the constraints of IoT environments, offering a balance between transmission efficiency, energy consumption, and image quality. The integration of chaotic sensing and hybrid reconstruction ensures that the system not only minimizes the amount of data transmitted but also maintains privacy and robustness against channel noise.

Algorithm 1: Image Sampling with Chaotic Compressed Sensing

Input: Grayscale image I of size (H, W) (8-bit)
Input: Chaotic control parameter r, initial value X_0, threshold $\in [0, 1]$
Output: Array y_{nonzero} of (index, value) pairs

$x \leftarrow \text{reshape}(I, N), \quad N \leftarrow H \times W$;
$X \leftarrow X_0$;
for $i \leftarrow 1$ **to** N **do**
  $X \leftarrow X/|X| - r \cdot X$; `// Signum chaotic map`
  chaotic_seq$[i] \leftarrow X$;
end
Normalize chaotic_seq to range $[0, 1]$;
for $i \leftarrow 1$ **to** N **do**
  if *chaotic_seq*$[i] >$ *threshold* **then**
    mask$[i] \leftarrow 1$;
  else
    mask$[i] \leftarrow 0$;
  end
end
for $i \leftarrow 1$ **to** N **do**
  $y[i] \leftarrow \text{mask}[i] \cdot x[i]$;
end
$y_{\text{nonzero}} \leftarrow \emptyset$;
for $i \leftarrow 1$ **to** N **do**
  if $y[i] \neq 0$ **then**
    Append $(i, y[i])$ to y_{nonzero};
  end
end
Transmit y_{nonzero} over SPI to LoRa transmitter;

Algorithm 2: Image Reconstruction

Input: Compressed data y_{nonzero} (index, value pairs), image size (H, W), chaotic map parameters r, X_0, threshold, initial threshold λ, decay factor λ_{decay}, number of iterations
Output: Reconstructed image x_{recon} of size (H, W)

$N \leftarrow H \times W$;
Initialize x_{full} as a zero vector of length N;
for $(i, val) \in y_{nonzero}$ **do**
  $x_{\text{full}}[i] \leftarrow$ val;
end
$y_{\text{reshaped}} \leftarrow \text{reshape}(x_{\text{full}}, H, W)$;
Step 1: Regenerate chaotic binary mask;
$X \leftarrow X_0$;
for $i \leftarrow 1$ **to** N **do**
  $X \leftarrow X/|X| - r \cdot X$; `// Signum chaotic map`
  chaotic_seq$[i] \leftarrow X$;
end
Normalize chaotic_seq to range $[0, 1]$;
for $i \leftarrow 1$ **to** N **do**
  if *chaotic_seq*$[i] >$ *threshold* **then**
    mask$[i] \leftarrow 1$;
  else
    mask$[i] \leftarrow 0$;
  end
end
mask $\leftarrow \text{reshape}(\text{mask}, H, W)$;
Step 2: Iterative reconstruction;
$x_{\text{hat}} \leftarrow y_{\text{reshaped}}$;
for $i \leftarrow 1$ **to** *iterations* **do**
  $(\text{flat}, \text{slices}) \leftarrow \text{WaveletTransform}(x_{\text{hat}})$;
  flat_thresh $\leftarrow \text{SoftThreshold}(\text{flat}, \lambda)$;
  $x_{\text{hat}} \leftarrow$ InverseWaveletTransform(flat_thresh, slices);
  for *pixel* (m, n) **do**
    if *mask*$[m, n] = 1$ **then**
      $x_{\text{hat}}[m, n] \leftarrow y_{\text{reshaped}}[m, n]$;
    end
  end
  $x_{\text{hat}} \leftarrow \text{TVDenoise}(x_{\text{hat}}, \text{weight} = 0.025)$;
  $\lambda \leftarrow \lambda \cdot \lambda_{\text{decay}}$;
end
Clip values in x_{hat} to range $[0, 1]$;
return $x_{\text{recon}} \leftarrow x_{\text{hat}}$;

A. Chaotic-Based Sensing Matrix Generation

The compressed sensing process relies on a pseudo-random measurement matrix that satisfies the Restricted Isometry Property (RIP) and ensures incoherence with the sparsifying basis. In order to meet these requirements while also provide lightweight encryption, this work employs a chaotic map-based approach to generate the sensing matrix. The signum chaotic map, which is sensitivity to initial conditions and ability to generate highly random-like sequences from deterministic equations [23], [24], is given by:

$$X_{n+1} = \text{sign}(X_n) - rX_n \tag{1}$$

where r is a chaos tuning parameter. The simple form of (1) can be considered by

$$X_{n+1} = \frac{X_n}{|X_n|} - rX_n \tag{2}$$

where, X_n is the state at iteration n, and r, is the control parameter that determines the chaotic behavior. For appropriate values of r, the system exhibits aperiodic, non-converging sequences, making it suitable for generating random measurement matrices with strong incoherence properties. The measurement matrix generation can be achieved, following this procedure:

- Select a suitable initial condition $X_0 \in \{-1, 1\}$ and a control parameter $r \in \mathbb{R}$ that ensures chaotic behavior ($X_0 = 0.1$ and $r = 1.99$ in this experiment). Both values can be utilized as secret keys for encryption purposes.
- Iterate the signum chaotic map to generate a 1D chaotic sequence $\{X_n\}$. The total number of iterations should be equal to the number of elements required in the measurement matrix, i.e. $M \times N$, where M is the number of measurements and N is the signal/image dimension after vectorization.
- Normalize the chaotic sequence and optionally apply

binarization (e.g. using sign function) or map it to a standard distribution (e.g. Gaussian or Bernoulli) to match common CS practices.

The use of the signum chaotic map introduces an intrinsic encryption feature, as the sensing matrix depends heavily on the initial value X_0 and parameter r. Any slight variation in these parameters leads to entirely different matrices, making it extremely difficult to reconstruct the original image without the correct key. Thus, the chaotic map not only supports compressed sensing but also secures the data at the acquisition stage without the need for additional encryption layers.



Fig. 2. Custom-designed embedded device of transmission node.

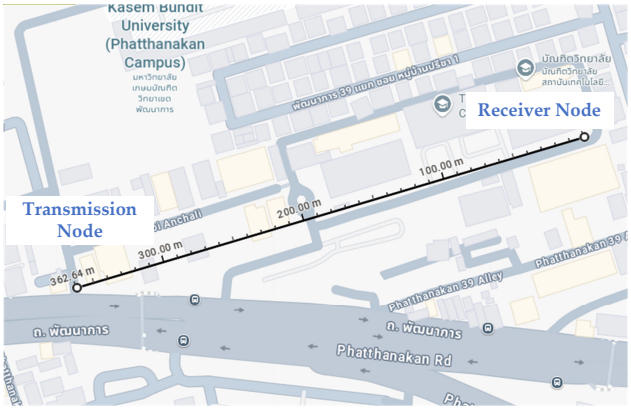


Fig. 3. Testing site of the proposed system.

TABLE I. SUMMARIZE OF LORA COMMUNICATION PARAMETER

Parameters	Value
LoRa Module	SX1262
Frequency	923 MHz
Spreading Factor	7
Coding Rate	4/5
Transmission Power	14 dBm

B. Image Sampling via Compressed Sensing

The image acquisition process is carried out using the principles of Compressed Sensing (CS), a signal processing technique that enables accurate recovery of sparse or compressible signals from a small number of linear measurements. This is particularly beneficial for image transmission in IoT systems, where bandwidth, memory, and energy are severely constrained. Let $x \in \mathbb{R}^N$ represent the original image signal in vectorized form, e.g. a grayscale image flattened into a 1D array. Under the CS framework, instead of acquiring the full signal x, a reduced number of linear measurements $y \in \mathbb{R}^M$ can be acquired with $M \ll N$ by projecting the signal onto a measurement matrix $\Phi \in \mathbb{R}^{M \times N}$. The data acquisition model can be expressed by:

$$y = \Phi x \tag{3}$$

In order to ensure that x can be reconstructed from y, the signal must be sparse in some transform domain. That is, there exists a sparsifying basis Ψ such that

$$x = \Psi s \tag{4}$$

where, s is a sparse coefficient vector. Substituting into the measurement equation yields:

$$y = \Phi \Psi s = \Theta s \tag{5}$$

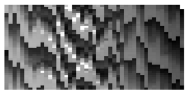
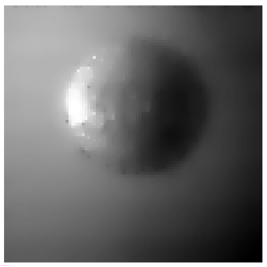
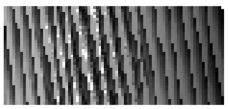
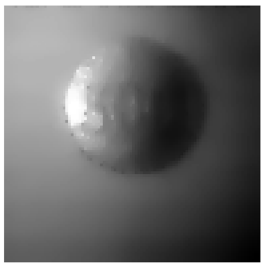
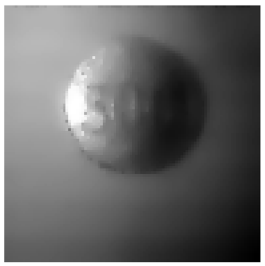
where, $\Theta = \Phi\Psi$ is the effective sensing matrix. The reconstruction of x involves recovering the sparse vector s from y and applying the inverse transform. In the proposed system, the compressed sensing matrix Φ is generated using signum chaotic map (as described in Section 2.1). This chaotic matrix introduces both measurement incoherence and data-level encryption. The image captured by the ESP32-CAM is first converted into grayscale and then vectorized into x. This vector is multiplied by Φ to produce the compressed measurements y, which are significantly smaller in size than the original image.

The sampling process begins with image acquisition from the camera module, followed by preprocessing steps that include grayscale conversion and normalization. The resulting image is then vectorized into a one-dimensional signal x, which is subsequently projected onto a chaotic map-generated sensing matrix Φ to compute the compressed measurements $y = \Phi x$. Finally, the compressed vector y is transmitted via the Serial Peripheral Interface (SPI) to the LoRa transmitter for wireless delivery. The pseudo-code of image acquisition and compressed sensing is illustrated in the Algorithm 1.

C. Reconstruction Algorithm

After compressed measurements are transmitted via LoRa and received by a computational node, the next step is to reconstruct the original image from the sparse measurement vector. The reconstruction process is designed to recover the original image from the received compressed measurements by leveraging the same chaotic sequence used during sampling.

TABLE II. PERFORMANCE EVALUATION OF COMPRESSED AND RECONSTRUCTED IMAGES OVER DIFFERENT COMPRESSION RATIOS

Original Image	CR	Compressed Image	Reconstructed Image	MSE	PSNR (dB)
	0.2			0.159	31.97
	0.3			0.123	32.92
	0.4			0.102	34.03

This approach ensures both spatial consistency and security by deterministically regenerating the sampling pattern.

First, the receiver regenerates the chaotic sequence (2) using the same initial condition and control parameter r as used during the sampling process. This sequence is normalized and binarized with a fixed threshold to reproduce the binary sampling mask, which distinguishes between known (non-zero) and unknown (zero) pixel locations. The received measurements are reshaped into the original image dimensions, and zero values are re-inserted at positions determined by the chaotic mask. This forms a sparse image matrix where only a subset of pixel values is available. The reconstruction process begins with this sparse image as the initial estimate and iteratively refines it using a combination of wavelet-domain sparsity enforcement and total variation (TV) denoising. Next, the reconstruction loop proceeds by transforming the current image estimate $\hat{\mathbf{x}}$ into the wavelet domain using a two-dimensional discrete wavelet transform, resulting in a sparse representation of the image coefficients. A soft-thresholding function is then applied to these coefficients to suppress noise and enforce sparsity. This function shrinks coefficients toward zero based on a threshold λ that decays over iterations as follows:

$$\tilde{X} = sign(X) \cdot max(|X| - \lambda, 0) \tag{6}$$

where, X and $\tilde{X}$ are the wavelet coefficient and the threshold coefficient, respectively. The threshold coefficients are subsequently transformed back into the spatial domain via the inverse wavelet transform, resulting in an updated image estimate. To ensure consistency with the original measurements, the known pixel values identified using the chaotic mask are reinserted into their respective positions. This operation can be mathematically expressed as:

$$\hat{x}^{(k)}[i] = \begin{cases} y[i], & \text{if } i \in \Omega. \\ \hat{x}^{(k)}[i], & \text{otherwise} \end{cases} \tag{7}$$

where, Ω is the set of known pixel indices from the chaotic sampling mask, and $\hat{x}^{(k)}$ is the reconstructed image at iteration k. To further refine the output, total variation denoising is applied to the reconstructed image to reduce artifacts and preserve edge structures. This can be formulated as the minimization of the TV norm, given by:

$$\min_x |||x||_{TV} \quad \text{subject to} \quad x \approx \tilde{x} \tag{8}$$

This iterative refinement is performed for 350 iterations. The threshold λ is initialized at 0.1 and decayed by a factor of 0.95 after each iteration to gradually enhance image details. Finally, the reconstructed image is clipped to the valid grayscale range $[0, 1]$. This hybrid reconstruction approach effectively balances sparsity promotion, noise suppression, and

data fidelity, allowing high-quality image recovery from highly compressed chaotic measurements. The pseudo-code of image reconstruction process is illustrated in the Algorithm 2.

III. Experimental Results

To validate the effectiveness of the proposed hybrid compressed sensing approach for image transmission over LoRa-based IoT networks, real-world experiments were conducted using ESP32-CAM as the image acquisition and compression node, and CubeCell AB01 as the LoRa communication. Fig. 2 illustrates the custom-designed embedded system used for field deployment of the proposed image transmission framework. These tests aimed to evaluate system performance in practical conditions, including image quality, transmission efficiency, and hardware feasibility in a low-power wireless environment.

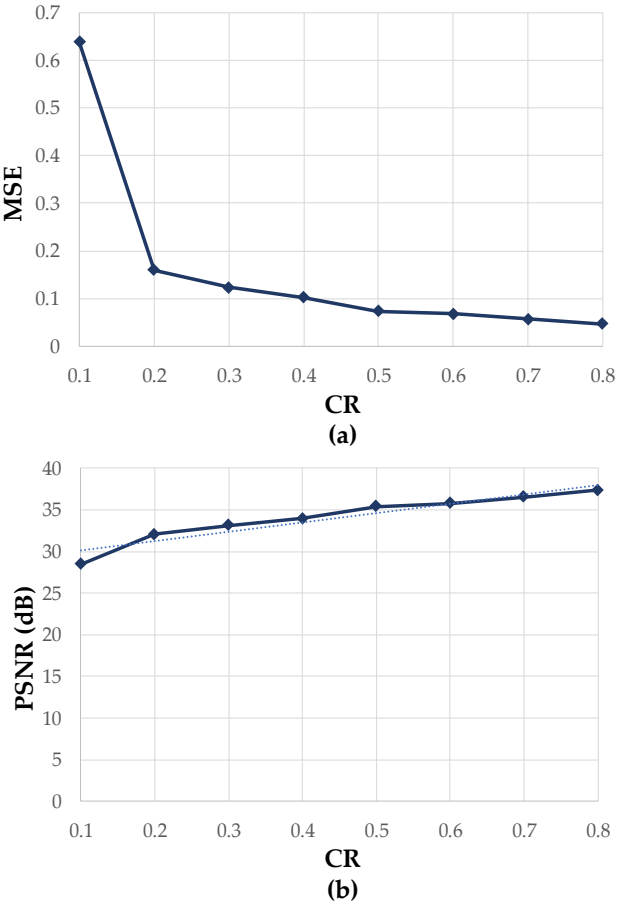


Fig. 4. Compression ratio vs (a) MSE and (b) PSNR.

A. Field Deployment

Fig. 3 shows the real-world testing site of the proposed image transmission system at the Thai-Nichi Institute of Technology (TNI) campus, specifically across the length of the Faculty of Engineering building. The devices were powered by portable USB battery packs and tested over a range of distances (up to 300 meters) with clear line-of-sight. The transmission path included partial obstructions such as walls, corridors, and concrete pillars, making it a meaningful scenario to assess LoRa signal reliability, data transmission performance, and image reconstruction quality over medium-range distances. Table I summarizes the LoRa parameters in the experiment.

B. Evaluation Metrics

This paper selects Mean Squared Error (MSE) as the primary loss function for training the reconstruction model. It measures the average squared difference between the reconstructed image and the original image, defined mathematically as:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}[y_i - f(x_i)]^2 \tag{9}$$

where, N represents the total number of image pixels (or samples), y_i is the ground truth pixel value, and $f(x_i)$ is the predicted pixel value. MSE serves as a direct indicator of reconstruction accuracy. Otherwise, the lower of MSE indicates the closer the reconstructed image is to the original.

To further assess the quality of the reconstructed images, Peak Signal-to-Noise Ratio (PSNR) is employed as quantitative evaluation metrics. PSNR is a widely used objective metric in image processing that reflects the ratio between the maximum possible pixel value and the magnitude of the reconstruction error, and is expressed in decibels (dB). It is calculated as:

$$\text{PSNR} = 10 \cdot \log_{10}(\frac{M^2}{\text{MSE}}) \tag{10}$$

where, M is the maximum possible pixel intensity value of the image. A higher PSNR indicates better reconstruction quality and lower image distortion, e.g. a PSNR value above 30 dB generally suggests that the reconstructed image is of good visual quality.

C. Performance Analysis

Table II presents a visual and quantitative analysis of the proposed reconstruction system under varying compression ratios (CR) of 0.2, 0.3, and 0.4. The original image is shown in the first column. The second column shows the encrypted or compressed image as produced by chaotic compressed sensing, appearing as random patterns with no visual resemblance to the original. The third column displays the reconstructed images after applying the wavelet-based soft-thresholding and total variation (TV) denoising algorithm. As the CR increases from 0.2 to 0.4, the quality of the reconstructed image significantly improves. This is further supported by the quantitative metrics in MSE and PSNR. At CR = 0.2, the MSE is 0.159 with a PSNR of 31.97 dB, while at CR = 0.4, the MSE reduces to 0.102 and the PSNR increases to 34.03 dB. These results confirm that higher compression ratios yield more accurate image reconstruction due to the availability of more non-zero measurements. However, even at low CR, the reconstruction preserves the essential features of the original image, demonstrating the robustness and effectiveness of the proposed chaotic compressed sensing and reconstruction framework.

TABLE III. COMPARISON WITH EXISTING SYSTEMS

Criteria	Edirisinghe et al. (2024) [25]	Kirichek et al. (2017) [26]	Wei et al. (2021) [27]	Guerra et al. (2023) [28]	**Proposed**
Technology	ESP32CAM, SX1278	Raspberry Pi Zero, SX1267	Raspberry pi 3B+, SX1276	Raspberry Pi 3B+, SX1272	ESP32CAM, SX1262
Compression Method	JPEG	JPEG/JPEG 2000	WebP with Base64	YCoCg, Wavelet Subbands	Compressed Sensing, Wavelet, TV Minimization
PSNR (dB)	-	23	33.84	21.5 to 27	32.92 (CR = 0.3)
Transmission Time (s)	20	-	25.7	90	30
Security	None	None	None	None	Chaotic Masking

Additionally, Fig. 4 depicts the relationship between MSE and PSNR as the compression ratio (CR) varies. These plot indicates that using more measurements results in a reconstructed image which is closer to the original in terms of pixel-wise accuracy. Conversely, the PSNR increases with higher CR, reflecting improved visual quality of the reconstructed image. It can be consider that, the proposed chaotic compressed sensing and wavelet-TV reconstruction framework achieves a good balance between compression and image quality. Notably, even at moderate compression ratios (CR = 0.4–0.5), the system maintains acceptable reconstruction performance, suggesting suitability for deployment in bandwidth-constrained LoRa-based IoT applications where transmission efficiency is critical.

Table III presents a comparative evaluation between the proposed method and several existing image transmission techniques over LoRa networks. The comparison is based on key parameters such as technology framework, compression method, image quality in terms of PSNR, transmission time, and security features. The proposed system employs the chaotic compressed sensing, wavelet transform, and TV minimization, which enable high efficiency image reconstruction with significantly reduced data volume. Notably, the proposed method achieves a PSNR of 32.92 dB at a compression ratio of 0.3, which is comparable to or better than other approaches, such as the 33.84 dB reported by Wei et al. (2021) using WebP compression, and considerably higher than the 23 dB and 21.5–27 dB reported by Kirichek et al. (2017) and Guerra et al. (2023), respectively. In terms of transmission time, the proposed method achieves a practical 30-second duration, which is faster than the 90 seconds required by Guerra et al., and close to the 25.7 seconds achieved by Wei et al., while also incorporating inherent chaotic masking for data security, unlike all compared works, which offer no security measures. Additionally, while most competing methods rely on image-centric compression formats like JPEG, WebP, or wavelet subbands, the proposed approach achieves compression and encryption simultaneously through chaotic sampling. Overall, this comparison highlights the efficiency, security, and image quality advantages of the proposed method, making it highly suitable for low-power, bandwidth-constrained image transmission in IoT scenarios.

IV. CONCLUSION

This paper presents a hybrid compressed sensing approach tailored for efficient and secure image transmission in LoRa-based IoT systems. The proposed method achieves both significant data compression and reliable image reconstruction by integrating a chaotic map-based sensing matrix with wavelet transforms and total variation (TV) minimization. The use of a chaotic sequence not only improves incoherence during the sampling process but also serves as a lightweight encryption mechanism, enhancing data privacy without additional computational overhead. Experimental evaluations on real embedded hardware demonstrate the practical feasibility of the proposed system in low-power environments. The results indicate that even with low compression ratios, the reconstructed images maintain high perceptual quality, with PSNR values exceeding 32 dB. Furthermore, transmission times are reduced to under 30 seconds per image, significantly outperforming traditional JPEG-based and wavelet-subband techniques used in LoRa imaging. A comparative analysis with existing methods highlights the advantages of the proposed approach in terms of image quality, transmission efficiency, power consumption, and embedded security. These benefits make the system particularly well-suited for IoT applications where bandwidth, energy, and processing resources are severely constrained. The proposed system has potential for a wide range of applications, including environmental monitoring, smart agriculture, structural health monitoring, and wildlife surveillance. It can also be applied for remote industrial inspections, where timely visual data from resource-constrained IoT nodes can significantly improve decision-making efficiency and operational safety.

ACKNOWLEDGMENT

The authors are grateful for the collaboration between Temasek Polytechnic and Thai-Nichi Institute of Technology.

REFERENCES

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things journal*, vol. 1, no. 1, pp. 22–32, 2014.

[2] R. Sanchez-Iborra, J. Sanchez-Gomez, J. Ballesta-Viñas, M.-D. Cano, and A. F. Skarmeta, "Performance evaluation of lora considering scenario conditions," *Sensors*, vol. 18, no. 3, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/3/772

[3] M. J. Faber, K. M. van der Zwaag, W. G. V. dos Santos, H. R. d. O. Rocha, M. E. V. Segatto, and J. A. L. Silva, "A theoretical and experimental evaluation on the performance of lora technology," *IEEE Sensors Journal*, vol. 20, no. 16, pp. 9480–9489, 2020.

[4] A. H. Jebril, A. Sali, A. Ismail, and M. F. A. Rasid, "Overcoming limitations of lora physical layer in image transmission," *Sensors*, vol. 18, no. 10, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/10/3257

[5] J. Kim, M. Kwak *et al.*, "Reliable image transmission over lora networks." *Issues in Information Systems*, vol. 25, no. 1, 2024.

[6] S. Li, H. Song, and M. Iqbal, "Privacy and security for resource-constrained iot devices and networks: Research challenges and opportunities," *Sensors*, vol. 19, no. 8, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/8/1935

[7] S. Kumar, D. Kumar, R. Dangi, G. Choudhary, N. Dragoni, and I. You, "A review of lightweight security and privacy for resource-constrained iot devices," *Computers, Materials and Continua*, vol. 78, no. 1, pp. 31–63, 2024.

[8] A. Shafique, A. Mehmood, M. Alawida, and A. N. Khan, "Enhancing privacy in data transmission between iot devices: A robust encryption and embedding framework for secure and meaningful image communication," *Journal of Information Security and Applications*, vol. 93, p. 104112, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214212625001498

[9] P. Hu, J. Im, Z. Asgar, and S. Katti, "Starfish: Resilient image compression for aiot cameras," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 395–408.

[10] P. Chakraborty, J. Cruz, and S. Bhunia, "Leveraging domain knowledge using machine learning for image compression in internet-of-things," *arXiv preprint arXiv:2009.06742*, 2020.

[11] A. Hojjat, J. Haberer, and O. Landsiedel, "Mcucoder: Adaptive bitrate learned video compression for iot devices," 2024. [Online]. Available: https://arxiv.org/abs/2411.19442

[12] Y. Shuling, Y. Renping, and W. Longzhi, "Transmission line monitoring technology based on compressed sensing wireless sensor network." *International Journal of Advanced Computer Science & Applications*, vol. 15, no. 4, 2024.

[13] H. Djelouat, A. Amira, and F. Bensaali, "Compressive sensing-based iot applications: A review," *Journal of Sensor and Actuator Networks*, vol. 7, no. 4, 2018. [Online]. Available: https://www.mdpi.com/2224-2708/7/4/45

[14] Z. Wang, S. Sun, Y. Li, Z. Yue, and Y. Ding, "Distributed compressive sensing for wireless signal transmission in structural health monitoring: An adaptive hierarchical bayesian model-based approach," *Sensors*, vol. 23, no. 12, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/12/5661

[15] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 83–91, 2008.

[16] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse mri: The application of compressed sensing for rapid mr imaging," *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[17] F. Chaparro B., M. Pérez, and D. Mendez, "A communication framework for image transmission through lpwan technology," *Electronics*, vol. 11, no. 11, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/11/1764

[18] L. R. Chandran, I. Karuppasamy, M. G. Nair, H. Sun, and P. K. Krishnakumari, "Compressive sensing in power engineering: A comprehensive survey of theory and applications, and a case study," *Journal of Sensor and Actuator Networks*, vol. 14, no. 2, 2025. [Online]. Available: https://www.mdpi.com/2224-2708/14/2/28

[19] M. S. Khan, A. Al-Dubai, J. Ahmad, N. Pitropakis, and B. Ghaleb, "A novel feature-aware chaotic image encryption scheme for data security and privacy in iot and edge networks," 2025. [Online]. Available: https://arxiv.org/abs/2505.00593

[20] E. Setyaningsih and R. Wardoyo, "Review of image compression and encryption techniques," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 2, 2017. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2017.080212

[21] L. Gong, K. Qiu, C. Deng, and N. Zhou, "An image compression and encryption algorithm based on chaotic system and compressive sensing," *Optics & Laser Technology*, vol. 115, pp. 257–267, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0030399218304961

[22] L. Li, Y. Fang, L. Liu, H. Peng, J. Kurths, and Y. Yang, "Overview of compressed sensing: Sensing model, reconstruction algorithm, and its applications," *Applied Sciences*, vol. 10, no. 17, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/17/5909

[23] W. San-Um and P. Ketthong, "The generalization of mathematically simple and robust chaotic maps with absolute value nonlinearity," in *TENCON 2014 - 2014 IEEE Region 10 Conference*, 2014, pp. 1–4.

[24] W. San-Um, P. Ketthong, W. Chankasame, and J. Noymanee, "A cost-effective true random bit generator using a pair of robust signum-based chaotic maps," in *2015 Science and Information Conference (SAI)*, 2015, pp. 1305–1310.

[25] S. Edirisinghe and Sachinda, "Image transmission using lora for edge learning," *SSRN Electronic Journal*, 02 2024.

[26] R. Kirichek, V.-D. Pham, A. Kolechkin, M. Al-Bahri, and A. Paramonov, "Transfer of multimedia data via lora," in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, O. Galinina, S. Andreev, S. Balandin, and Y. Koucheryavy, Eds. Cham: Springer International Publishing, 2017, pp. 708–720.

[27] C.-C. Wei, P.-Y. Su, C.-C. Chang, and K.-C. Chang, "A study on lora dynamic image transmission," in *2021 IEEE 4th International Conference on Knowledge Innovation and Invention (ICKII)*, 2021, pp. 10–13.

[28] K. Guerra, J. Casavilca, S. Huamán, L. López, A. Sanchez, and G. Kemper, "A low-rate encoder for image transmission using lora communication modules," *International Journal of Information Technology*, vol. 15, no. 2, pp. 1069–1079, Feb 2023. [Online]. Available: https://doi.org/10.1007/s41870-022-01077-7

Predictive Models in Mental Health Based on Unsupervised Data Clustering

Inoc Rubio Paucar[1], Cesar Yactayo-Arias[2], Laberiano Andrade-Arenas[3]
Facultad de Ingeniería y Negocios, Universidad Privada Norbert Wiener, Lima, Perú[1]
Departamento de Estudios Generales, Universidad Continental, Lima, Perú[2]
Facultad de Ciencias e Ingeniería, Universidad de Ciencias y Humanidades, Lima, Perú[3]

***Abstract*—In the university context, students' mental health has been progressively affected over time. The objective of this research was to develop a predictive model of machine learning based on the K-Means algorithm, with the purpose of identifying and classifying mental health profiles among university students. For the construction of this model, the standard Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology was applied, which encompasses five stages: business understanding, data understanding, data preparation, modeling, and evaluation. The results obtained suggest that the generated clusters produce consistent groupings in key variables such as screen time, hours of sleep, and level of physical activity, allowing the characterization of different student profiles. This approach provides valuable information for designing academic support strategies and programs aimed at students' well-being and mental health. The early identification of behavioral patterns and lifestyle habits enables educational institutions to implement preventive and personalized measures, fostering improved academic performance and university adaptation.**

***Keywords*—Behavioral patterns; clustering; machine learning; mental health; university students**

I. Introduction

In the field of higher education, mental health has emerged as a critical issue in recent years due to the notable increase in disorders such as anxiety, depression, and chronic stress. University students constitute a particularly vulnerable population, as they face multiple challenges inherent to this formative stage, including academic pressure, adaptation to new environments, uncertainty regarding professional futures, financial difficulties, and, frequently, separation from family support networks. According to the World Health Organization (WHO), approximately 20% of young people experience some type of mental disorder, many of which begin precisely during university life [1] [2]. In this respect, it becomes evident that this problem demands attention in every social context, as it exacerbates personal difficulties among university students worldwide.

Mental health in the university population represents both a scientific and medical issue of growing relevance [3]. From a medical perspective, mental disorders such as anxiety, depression, and chronic stress not only compromise students' emotional well-being but also directly affect their physical health, quality of life, and academic performance. If not detected or treated in time, these disorders may lead to serious consequences such as university dropout, deterioration of social relationships, and, in the most severe cases, suicide attempts. From a scientific perspective, this issue poses significant challenges for interdisciplinary research [4] [2]. Experts in medicine consider mental health problems as a set of conditions that comprehensively affect student life, also impacting the family environment.

There is an urgent need to better understand the individual, social, and contextual risk factors that contribute to the onset of these disorders during university years. Despite the rising prevalence, many cases remain underdiagnosed due to stigmatization, lack of institutional resources, or limited awareness of the importance of preventive psychological care. Furthermore, the scientific study of student mental health requires the development and validation of effective tools for early diagnosis, symptom monitoring, and implementation of personalized interventions. The integration of approaches based on emerging technologies, such as artificial intelligence and machine learning, opens new possibilities for risk detection and prediction; however, gaps in the evidence remain and must be addressed through rigorous research [5] [6]. Technologies thus represent a viable option within the broader social challenge, constituting an impactful factor for prevention depending on the context under study.

This research is justified by the need to reduce the incidence of psychological disorders and other problems that significantly affect university students, negatively impacting the achievement of their academic goals and their professional future. In this regard, it is essential to develop effective mechanisms to counteract these difficulties, thereby promoting students' overall well-being and supporting their persistence and success in higher education. Faced with this issue, it becomes crucial to implement efficient mechanisms that enable the timely identification of at-risk students in order to design and implement personalized intervention strategies [7] [8]. In this context, the use of machine learning techniques represents an innovative and powerful tool, as it allows the analysis of large volumes of data and the discovery of hidden patterns that would not be detectable through traditional methods. Specifically, this research proposes the use of clustering algorithms, such as K-means, to group students according to relevant psychological, academic, and social characteristics. This methodology will allow the segmentation of the student population into groups with similar risk levels, facilitating informed decision-making by educational institutions [9]. In this way, more effective support strategies can be developed, aimed at improving mental health, student retention, and academic success, thus contributing to students' overall well-being and to the strengthening of the higher education system.

The objective of this research is to develop a predictive model based on clustering techniques and machine learning, aimed at identifying and classifying mental health profiles among university students through the analysis of multivariate data related to lifestyle habits, academic characteristics, and emotional factors. The purpose of this model is to facilitate the early detection of potential psychological disorders and to contribute to the design of preventive strategies or personalized interventions that enhance student well-being and academic performance.

II. Literature Review

This section presents the structure of the literature review, which is divided into two main parts. The first addresses the theoretical foundations, providing a detailed discussion of the mental health variable and the clustering approach, with emphasis on the K-means algorithm. The second part comprises the related works, analyzing previous studies conducted by various authors who have applied machine learning techniques, specifically clustering algorithms, to the study and analysis of mental health among university students. The purpose of this review is to provide the theoretical basis for the research and to highlight the relevance and applicability of the proposed approach.

A. Theoretical Bases

1) Mental health: Mental health in university students refers to the emotional, psychological, and social state that directly influences their overall well-being and academic performance. This stage represents a critical period in personal development, where students face multiple challenges such as academic pressure, adaptation to new environments, family expectations, and, in many cases, financial limitations. In particular, some students must finance their studies without family support, which significantly increases their emotional and financial burden [10] [11]. This situation of solitude and economic responsibility can generate high levels of stress and anxiety, affecting their mental health and hindering the achievement of their academic goals. Table I presents the key factors influencing the mental health of university students, detailing the causes and the associated emotional and psychological effects, such as stress, anxiety, and feelings of isolation, which may impact their well-being and academic performance.

2) Clustering: Clustering is an unsupervised data analysis technique aimed at identifying and grouping objects or instances into subsets known as clusters. Its fundamental principle is that elements within the same group share a high degree of internal similarity, while exhibiting marked differences from elements in other groups. The evaluation of this similarity or distance is carried out using various metrics, including Euclidean distance, Manhattan distance, and correlation coefficients, selected according to the characteristics and nature of the data. Unlike supervised methods, clustering does not require predefined labels, which makes it a powerful exploratory tool for uncovering hidden patterns, latent structures, and intrinsic relationships in large volumes of information [12] [13]. Its versatility has positioned it as an essential resource in areas such as market segmentation, bioinformatics, image analysis, and anomaly detection. Table II shows the representation of three metrics for evaluating clusters: Euclidean Distance measures the closeness between points; the Silhouette Score (0.65) indicates good group separation; and the Calinski–Harabasz Index (200) reflects well-defined and separated clusters.

B. Related Work

Several studies have indicated that multiple factors related to university students' mental health can lead to different illnesses. The authors [14] [15] highlight the growing concern regarding mental health problems in university students. In particular, Singh evaluated various machine learning techniques, such as classification methods, clustering, and a hybrid neural network, applied to an academic dataset from the engineering faculty. The results showed that the hybrid algorithm, by combining classification and clustering approaches, significantly outperformed the other models, achieving high accuracy in predicting both academic performance and students' mental well-being. Unlike the previous study, this author addressed the influence of mental, emotional, social, and spiritual health on the academic performance of university students. To this end, data were collected through a questionnaire administered to 214 undergraduate students, and machine learning techniques such as feature selection, regression, neural networks, Naïve Bayes, and multidimensional analysis were employed, along with tools such as SPSS, R, and Python. The analysis revealed that these methods were effective in predicting academic performance and identifying the psychosocial factors with the greatest impact during the transition to virtual and hybrid learning modalities. On the other hand, the author [16] proposed an intervention program aimed at improving the mental health of university students, using a Bayesian model for the dynamic updating of parameters. Cluster random sampling was also used to divide participants into an experimental group and a control group. The results showed that the intervention produced positive changes in key aspects such as mental health literacy, self-efficacy in coping with psychological problems, and behavioral attitudes toward seeking help. In another study, an author developed a strategy to classify stress in engineering students using wearable devices that recorded signals during the Montreal Imaging Stress Task (MIST). Using machine learning models, the study managed to classify stress as resting, moderate, and high, achieving accuracies of 99.98% with EEG signals and 99.51% with Electrodermal Activity (EDA), Heart Rate (HR), and Skin Temperature (SKT), outperforming previous studies and demonstrating the effectiveness of these devices for real-time monitoring [17].

In a relevant study on university students' mental health, anxiety levels and their causes were investigated using a Likert-scale-based questionnaire administered to 127 participants. Based on the collected data, machine learning models such as Naïve Bayes, Decision Tree, Random Forest, and Support Vector Machine (SVM) were trained. The results were promising, with accuracies of 71.05% for Naïve Bayes and Decision Tree, 78.9% for Random Forest, and 75.5% for SVM, showing the potential of these algorithms to effectively predict anxiety levels and provide valuable insights into their causes. In another investigation, the application of predictive models for addressing mental health issues

TABLE I. FACTORS AFFECTING THE MENTAL HEALTH OF UNIVERSITY STUDENTS

Factor	Description	Impact on mental health
Academic pressure	Demands and high expectations in school performance	Stress, anxiety, exhaustion
Adaptation to new environments	Changes in social life and university environment	Loneliness, uncertainty, anxiety
Family expectations	Pressure to meet family goals	Anxiety, fear of failure
Economic limitations	Difficulty in affording studies without family support	Financial stress, constant worry
Individual responsibility	Emotional and financial burden of handling everything alone	Increased stress, feeling of isolation

TABLE II. COMMON METRICS TO EVALUATE CLUSTERING QUALITY

Metric	Description	Example value
Euclidean Distance	Measures the direct distance between points	-
Silhouette Score	Quality of clustering (from -1 to 1)	0.65
Calinski-Harabasz Index	Separation between clusters and internal cohesion	200

in university students employed approaches such as neural networks and support vector machines. These models were key in documenting a bibliometric analysis, whose results revealed that Artificial Intelligence and Machine Learning show a consistent growth trend in addressing mental health problems among young university students [18] [19]. The lack of timely attention to mental health problems increases the risk of worsening over time. This study collected data through questionnaires on depressive, anxiety, and obsessive-compulsive symptoms, applying models such as multiple linear regression to identify associations and the random forest algorithm for predictions. The results indicated that 20% of students showed severe depression and/or suicidal ideation, that financial concerns were strongly associated with depression, and that random forest showed high accuracy in predicting the maintenance of well-being (balanced accuracy = 0.85) and absence of suicidal ideation, but low accuracy in detecting worsening cases (balanced accuracy = 0.49). Furthermore, in another study on mental health in university students, it was identified that cognitive and somatic symptoms of depression were the most influential variables, with a high negative predictive value (0.89) and practically null positive predictive value, highlighting stress as a key parameter for mental well-being. Through surveys administered to 144 students from various universities, multilayer perceptrons (MLP) combined with Principal Component Analysis (PCA) and Grid Search Cross-Validation (GSCV) were used, achieving an accuracy of 80.5%, precision of 1.000, F1-score of 0.890, and recall of 0.826. The findings revealed that 11.26% presented high social stress and 24.10% extremely high psychological stress, although the study was limited by the use of non-probabilistic sampling and self-reported data [20] [21].

On the other hand, stress is a key factor in understanding the prevalence of mental health problems, which is why this study applied automated machine learning methods to identify its main causes. A survey was conducted with 355 students from 28 universities, using the Boruta algorithm to select the most relevant predictive factors. The models evaluated included decision tree, random forest, and support vector machine, and their performance was measured using confusion matrix, ROC curves, and k-fold cross-validation. The results highlighted that heart rate, systolic and diastolic blood pressure, sleep quality, smoking habits, and academic history were the most influential variables in predicting stress, with the random forest model achieving the best performance, reaching an accuracy of 89.72% and an area under the curve (AUC) of 0.8715, outperforming the other techniques in accuracy and predictive capacity. An important factor is social media, which is a determinant of mental health; thus, the author proposed evaluating its influence through the collection and analysis of 66,000 student posts. The research focused on the use of natural language processing models to identify symptomatic expressions of depression, anxiety, and stress, also employing the SARIMA algorithm to forecast the number of mental health consultations on campus. The incorporation of social media data significantly improved the accuracy of these predictions, achieving a correlation of r = 0.86 and a symmetric mean absolute percentage error (SMAPE) of 13.30, representing a 41% improvement compared to models that did not consider social information, thereby demonstrating the key value of social media in detecting and monitoring student mental well-being [22] [23]. Another study conducted with university students using multicenter cross-sectional surveys aimed to compare the performance of different machine learning models, including k-nearest neighbors, naïve Bayes, neural networks, and random forest. The results showed that algorithms based on Random Forest achieved the highest accuracy in identifying negative indicators of mental well-being. Among the most relevant variables for prediction were the number of weekly sports activities, body mass index, academic average, sedentary hours, and age. These findings provide valuable recommendations for modernizing and optimizing mental health assessment and monitoring, both at the individual level and within the university environment. In relation to the previous study, the research developed the Emotion Based Mental Health Classifier (EMHC), a system that combines sentiment analysis and categorical data to classify students' mental health into three levels: good, moderate, and poor. Based on data from a web application applied in universities in the NCR region of India, it uses advanced algorithms such as DeepFace, K-means, and Support Vector Classifier. This innovative tool facilitates the assessment and monitoring of student mental well-being in a global context of increasing psychological burden [24] [25].

In contrast to other studies, a multidimensional feedback

approach was proposed based on a clustering analysis applied to data from 174 university students. Using the k-means algorithm, participants without severe symptoms were segmented into three differentiated groups, while those with severe symptoms received professional care recommendations. This personalized feedback not only improved mental health literacy but also fostered a sense of belonging, reducing resistance to seeking support and enabling timely intervention [26]. Another important factor regarding alcohol consumption and perceived stress in Ecuadorian university students was analyzed using 7,134 records obtained from the Alcohol Use Disorders Identification Test (AUDIT-C) questionnaire, the PSS-10 scale, and sociodemographic data. By applying Lanz's (2013) methodology and using k-means and hierarchical clustering algorithms, three distinct groups were identified that reflect combined patterns of alcohol consumption and stress levels, thus supporting the theory linking both factors and providing a complementary perspective to traditional statistical analysis. Finally, another study sought to raise awareness of the mental health status of students at Satya Wacana Christian University, using the K-Means machine learning algorithm to identify emotional patterns based on numerical responses from 32 students. The analysis, conducted with Orange3 and based on the silhouette index, formed three groups: one associated with depression, mainly including students from the 2018 cohort and some from 2020; another group reflecting mental prosperity, composed of students from the 2018, 2019, and 2020 cohorts; and a third group in a state of harmony, with students from various cohorts between 2017 and 2019. The results show that the year of admission influences mental well-being, with greater prosperity observed in earlier cohorts, higher depression among third-year students, and a group that achieves emotional balance across different cohorts [27] [28]. In summary, related work presents important limitations that open up opportunities for further research: most are based on unrepresentative samples, which restricts the generalizability of the findings; these could be improved with more representative data and the best use of more advanced techniques.

III. Methodology

A. Definition of the CRISP-DM Methodology

CRISP-DM methodology is a widely used standard for guiding data mining and machine learning projects. Its purpose is to provide a structured, flexible, and non-linear framework that enables the development of models and the extraction of useful knowledge from data. This approach is characterized by being iterative and adaptable, meaning that its phases can be repeated or feed back into one another. In this way, the process can be continuously redesigned to respond to changing business needs, ensuring that solutions generate tangible value aligned with the strategic objectives of the organization [29] [30]. Additionally, Fig. 1 illustrates the CRISP-DM methodology for data analysis, which begins with information collection and preparation, followed by problem definition and data understanding using descriptive statistics. Variables are subsequently transformed for cluster modeling (K-Means), evaluating metrics such as silhouette and inertia, and finally, the resulting clusters are interpreted and validated. The process defined by this methodology represents a flexible form of implementation in machine learning projects, allowing its adaptation according to the specific topic and objectives of each case. Fig. 2 illustrates the architecture of the methodology and internally describes each process, showing how the model is developed according to the technical specifications.

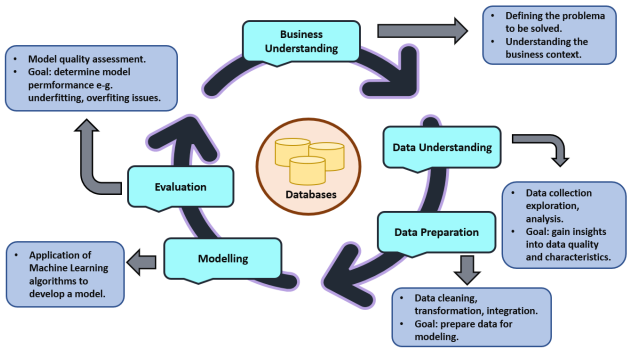


Fig. 1. CRISP-DM methodology.

1) Business understanding: In this section, corresponding to the first stage of the CRISP-DM methodology, the specific criteria necessary to address the problem are analyzed. Based on the data obtained, the problem to be solved is posed, which involved abstracting the real situation and transforming it into a clear and achievable analysis [31]. According to Table III, information is presented from a database downloaded from Kaggle, organized into records containing variables related to individuals' personal characteristics and habits. Fields include data such as gender, age, weight, height, type of transportation used, consumption of fast food and alcoholic beverages, among other factors. This information is valuable because it allows for the analysis of behavioral patterns linked to nutrition, physical activity, and lifestyles, which can be used in public health studies, obesity prediction, or in the development of preventive strategies to improve the population's quality of life.

2) Data understanding: For this section, certain data analysis concepts were applied using the Python programming language, which allowed us to examine and explore the important variables for the proposed model to understand their importance, detect possible inconsistencies and analyze the relationship between variables, serving as a fundamental basis for preparing the data that will feed the algorithmic model [32]. In this regard, Table IV shows a statistical summary of the key variables analyzed in the sample of 1,000 students. Demographic data, lifestyle habits, and perceived stress levels are included. The average age of participants is 20.34 years, with values ranging from 15 to 26. On average, students spend 6.91 hours in front of screens per day, sleep 6.45 hours, and engage in 5.02 hours of physical activity per week. The stress level, coded from 1 (low) to 3 (high), averages 1.85, suggesting that most students fall between low and medium levels of stress. This summary provides an understanding of the distribution and variability of the variables studied, serving as a reference for subsequent analysis and the development of predictive models.

On the other hand, Fig. 3 of boxplots presents a summary view of the distribution of four key variables: screen time, sleep duration, physical activity, and stress level. Each box shows the interquartile range, with the median dividing the distribution in half. The whiskers extend to non-outliers, and individual

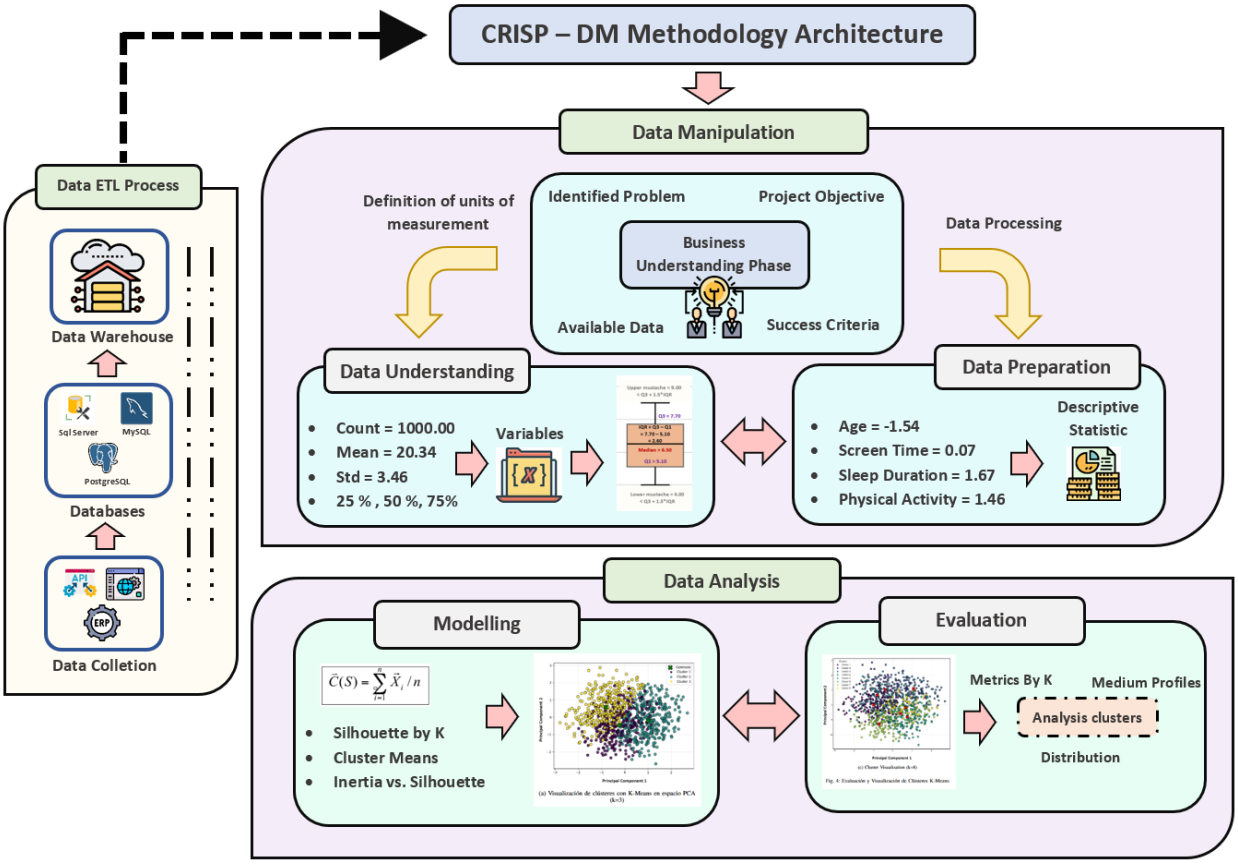


Fig. 2. Architecture CRISP-DM methodology.

TABLE III. BUSINESS UNDERSTANDING PHASE

Element	Description
Identified Problem	There are increasing mental health issues among university and high school students, reflected in factors such as stress, irregular sleep, excessive screen time, and low physical activity. This affects academic performance and overall well-being.
Project Objective	Use the information from the dataset (lifestyle habits, academic and emotional variables) to develop a predictive model based on clustering and machine learning techniques, which allows identifying students at risk or with similar mental health patterns.
Available Data	Demographics: name, gender, age, educational level.
	Lifestyle habits: screen time, sleep duration, weekly physical activity.
	Emotional factors: stress level, anxiety before exams.
	Observed outcome: change in academic performance.
Success Criteria	**Technical:** achieve well-defined and stable clusters (metrics such as Silhouette Score or Davies-Bouldin Index).
	Practical: obtain interpretable profiles that allow designing early intervention strategies or academic and emotional support.

TABLE IV. STATISTICAL SUMMARY OF THE STUDY VARIABLES

Stadistic	Age	Screen Time (hrs/día)	Sleep Duration (hrs)	Physical Activity (hrs/Week)	Stress Level
count	1000.00	1000.00	1000.00	1000.00	1000.00
mean	20.34	6.91	6.45	5.02	1.85
std	3.46	2.91	1.47	2.93	0.70
min	15.00	2.00	4.00	0.00	1.00
25%	17.00	4.40	5.10	2.60	1.00
50%	20.00	6.90	6.50	5.00	2.00
75%	23.00	9.50	7.70	7.60	2.00
max	26.00	12.00	9.00	10.00	3.00

points beyond them represent outliers. This graph reveals that all four variables exhibit data dispersion, with the presence of outliers that may indicate extreme behaviors or measurements in the studied sample, which warrants further investigation to

understand why these data deviate from the main behavior.

3) Data preparation: According to this section, in the Data Preparation phase, raw data is transformed into high-quality data ready for modeling. This stage, which goes hand in hand with the Data Understanding phase, involves cleaning the data (such as handling missing and outlier values) and transforming categorical variables into numerical variables so that the model's algorithms can properly process them [33].

Table V presents a sample of student data with variables normalized for analysis. Each row represents a student, and the columns display their data relative to the average. Age, Screen Time, Sleep Duration, and Physical Activity are standardized values, where a positive value indicates the student has an above-average age, screen time, sleep duration, or physical activity, and a negative value indicates a lower one. Stress Level is categorized qualitatively (Low, Medium, High) and also represented numerically (Stress Level Num) for use in machine learning models. For example, the first student has a below-average age (-1.54), above-average sleep duration (1.67), and a medium stress level. Table VI summarizes the descriptive statistics for the variables in a dataset after normalization. Numerical variables such as Age, Screen Time, Sleep Duration, and Physical Activity now have a mean of 0 and a standard deviation of 1, indicating they have been standardized. Meanwhile, the categorical variable Stress Level was converted to the numerical variable Stress Level Num, with a mean of 2.16, showing that the data tend toward the "Medium" stress level.

4) Modelling: In this section, the proposed algorithmic model is developed to address the identified problem. Supported by the analysis carried out in the previous phases, statistical graphs, tables, and mathematical expressions are defined to justify the selection and application of the K-Means algorithm as a clustering technique [34].

Fig. 4a shows the two-dimensional projection of the students through a Principal Component Analysis (PCA), where the three clusters identified by the K-Means algorithm are represented. Each color corresponds to a different group, and the centroids are indicated with 'X' markers. This visualization allows for the appreciation of the relative separation between groups and the distribution of the data in the reduced space. On the other hand, Fig. 4b presents the variation of the Silhouette index for different values of K (number of clusters). The maximum point is reached at k = best k, indicating the best partition in terms of internal cohesion and separation between groups. Additionally, the mathematical formula of the Silhouette coefficient is included, which reinforces the interpretation of the criterion used to validate the optimal number of clusters.

As shown in Table VII, the clustering analysis provides a clear overview of the data distribution and group characteristics. Table (a) reports the Silhouette Scores for different values of k, highlighting that the clustering quality is highest at $k = 3$ with a score of 0.51. Table (b) presents the mean values of Screen Time, Sleep Duration, and Physical Activity within each cluster, allowing us to profile the groups: Cluster 0 exhibits moderate screen time and sleep, Cluster 1 shows lower screen time and higher physical activity, and Cluster 2 has higher screen time with lower physical activity. Finally, Table (c) summarizes the inertia values along with the Silhouette Scores for each k, providing insight into both the compactness and separation of the clusters. Overall, these tables collectively illustrate the optimal number of clusters and the characteristics of each cluster group.

B. K-means Mathematical Formulas

1) Distance to centroid: The distance between a point x_p and a cluster centroid μ_i is computed using Euclidean distance (see eq.1) [35].
This distance is used to assign points to the nearest cluster.

$$d(x_p, \mu_i) = \sqrt{\sum_{j=1}^{m} (x_{pj} - \mu_{ij})^2} \tag{1}$$

where, m is the number of dimensions. This distance is used to assign points to the nearest cluster and to compute intra-cluster cohesion.

2) Cluster assignment: Each point is assigned to the cluster with the nearest centroid (see Eq. 2).
Points are grouped with the closest centroid to ensure cluster cohesion [36].

$$C_i = \{x_p : \|x_p - \mu_i\|^2 \leq \|x_p - \mu_j\|^2, \forall j = 1, \ldots, k\} \tag{2}$$

where, C_i is the set of points in cluster i. This ensures that points are grouped with the closest centroid.

3) Centroid update: Centroids are recalculated as the mean of all points in the cluster (see Eq. 3).
This moves the centroid to the center of its cluster for better stability.

$$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x \tag{3}$$

This adjustment moves the centroid to the center of its cluster, improving cluster cohesion.

4) Cost function (Inertia): K-means minimizes the sum of squared distances from points to their centroids (see Eq. 4).
Lower inertia indicates tighter clusters and better cluster compactness.

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{4}$$

Lower inertia means points are closer to their centroid, indicating tighter clusters.

5) Convergence criterion: The algorithm stops when centroids change minimally between iterations (see Eq. 5).
A small tolerance ϵ ensures that clusters are stable.

$$\|\mu_i^{(t+1)} - \mu_i^{(t)}\| < \epsilon, \quad \forall i \tag{5}$$

where ϵ is a small tolerance value. This ensures clusters are stable.

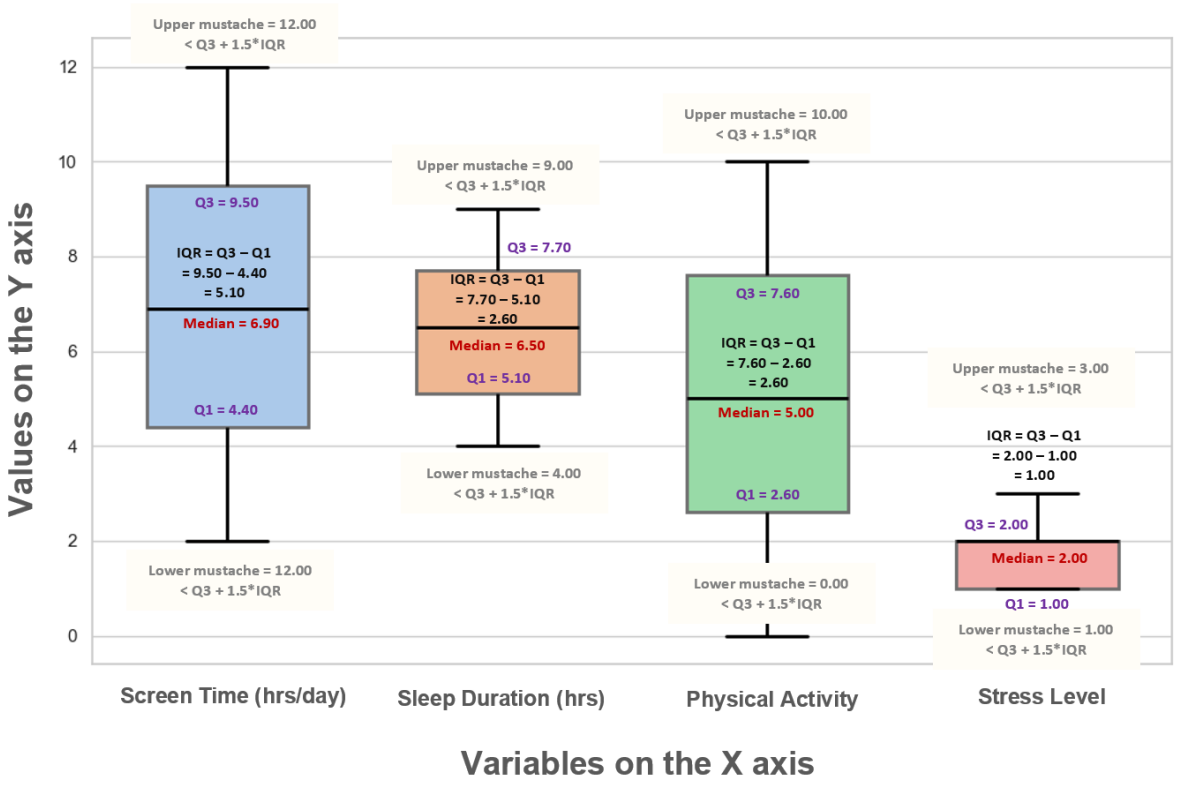


Fig. 3. Distribution of variables and the presence of outliers.

TABLE V. STUDENT DATA SAMPLE

Age	Screen Time (hrs)	Sleep Duration (hrs/week)	Physical Activity	Stress Level	Stress Level Num	Academic Performance	Anxiety Before Exams
-1.54	0.07	1.67	1.46	Medium	2.00	3.5	B
1.35	-1.24	-0.99	-1.64	Medium	2.00	2.8	C
-0.10	0.89	-0.72	0.40	Medium	2.00	3.2	B
-0.10	1.34	-0.58	0.16	High	3.00	2.1	D
-0.97	-1.41	-0.72	-0.65	Medium	2.00	3.9	A
0.50	0.20	-0.10	0.50	Low	1.00	4.1	A

TABLE VI. DESCRIPTIVE STATISTICS OF NUMERICAL VARIABLES

Variable	Mean	Standard Deviation	Minimum	Maximum
Age (Normalized)	0.00	1.00	-1.54	1.35
Screen Time (Normalized)	0.00	1.00	-1.41	1.34
Sleep Duration (Normalized)	0.00	1.00	-0.99	1.67
Physical Activity (Normalized)	0.00	1.00	-1.64	1.46
Stress Level Num	2.16	0.72	1.00	3.00

TABLE VII. (A) SILHOUETTE SCORES BY k, (B) CLUSTER MEANS, AND (C) INERTIA VS. SILHOUETTE

(a) Silhouette Scores by k

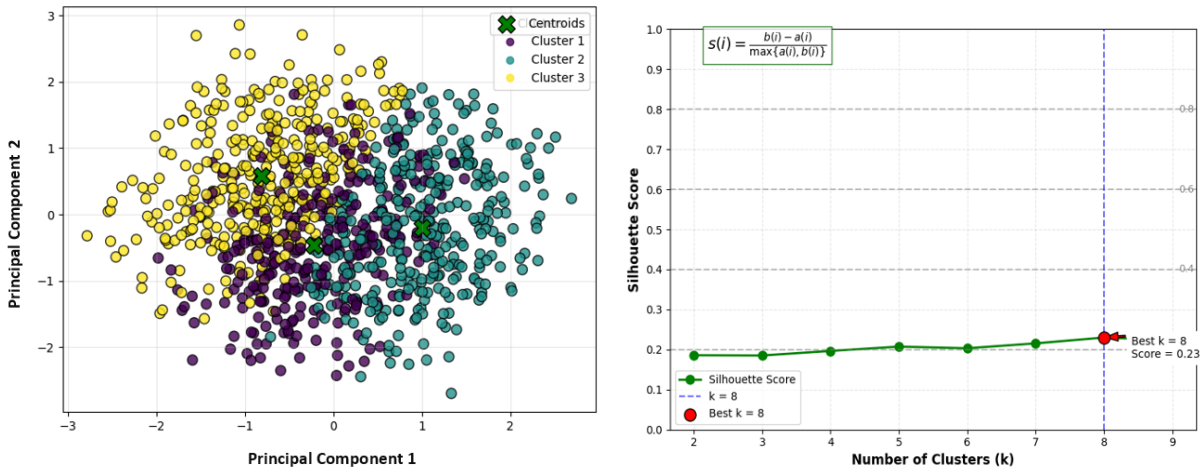
k	Silhouette
2	0.42
3	0.51
4	0.47
5	0.45
6	0.43

(b) Cluster Means (Example)

Cl.	Screen	Sleep	Activity
0	5.3	6.8	2.1
1	3.1	7.5	3.8
2	6.4	5.2	1.7

(c) Inertia vs. Silhouette

k	Inertia	Silhouette
2	580.1	0.42
3	430.4	0.51
4	390.2	0.47
5	350.1	0.45
6	320.5	0.43



(a) Visualizing clusters with K-Means in PCA space (k=3).

(b) Evaluating the optimal number of clusters with silhouette score.

Fig. 4. Student clustering using K-Means: PCA visualization and analysis of the optimal number of groups.

6) Total inertia (Optional): The total inertia quantifies overall cluster compactness (see Eq. 6).

It measures how well points fit within their clusters; lower values indicate tighter clustering.

$$J_{total} = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{6}$$

where, k is the number of clusters, C_i the points in cluster i, x a data vector, and μ_i the cluster centroid.

7) Cluster prediction: Once the algorithm converges, a new point x_{new} is assigned to the cluster with the nearest centroid.

This completes the clustering process, assigning each point to its optimal cluster (see Eq. 7).

$$\hat{C} = \arg\min_i \|x_{\text{new}} - \mu_i\| \tag{7}$$

This completes the K-means clustering process, assigning each point to its optimal cluster.

IV. Result

In the Evaluation phase of the CRISP-DM process, the quality of the obtained model is verified, comparing its results with the business objectives and criteria defined in the initial phases. This process not only involves analyzing statistical and performance metrics, but also evaluating the consistency of the clusters generated by the K-Means algorithm with respect to the problem context. In this way, it is determined whether the model is suitable for practical implementation or if it requires additional adjustments to the parameters, input data, or the methodology used [37] [38].

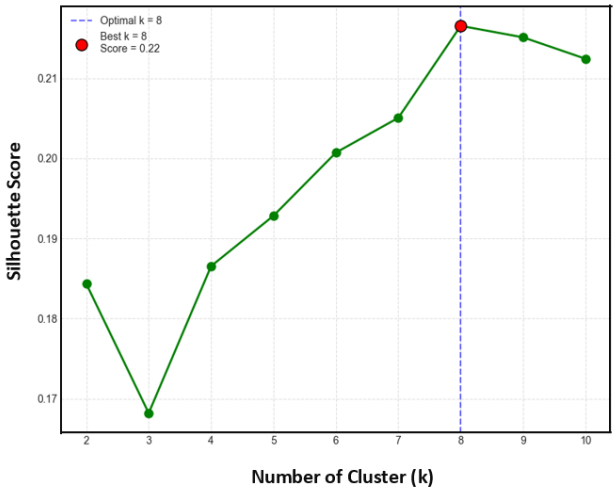
A. Evaluation of Result

Fig. 5 presents a comprehensive evaluation of the K-Means clustering analysis. Fig. 5a illustrates the Inertia curve (elbow method), which is used to identify the optimal number of clusters (k) by minimizing the sum of squared distances from the points to their centroids. Fig. 5b complements this evaluation with the Silhouette Score, a metric that measures cluster cohesion, where higher values indicate that objects are well grouped. Finally, Fig. 5c provides a visualization of the resulting clusters in a PCA (Principal Component Analysis) space, allowing observation of both the data distribution and the location of centroids for each selected cluster.

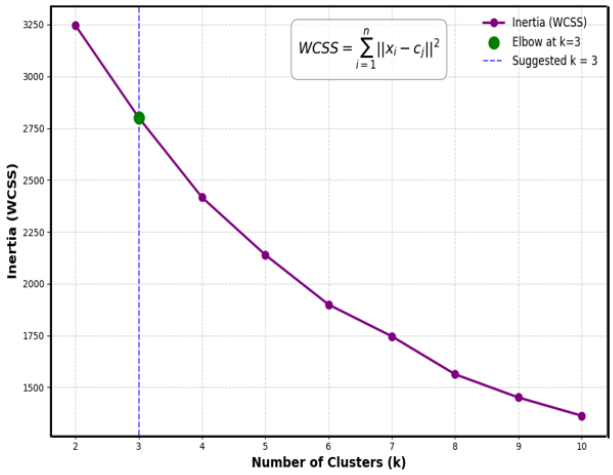
The references in the group of Tables VIII provide a comprehensive overview of the K-Means cluster analysis. Table VIIIa of Metrics by k shows that the Silhouette Score reaches its highest point at $k = 8$, suggesting that this is the optimal number of clusters for segmenting the data. Complementing this finding, Table VIIIb of Cluster Distribution demonstrates that the resulting 8 clusters have a relatively balanced size, with no groups being excessively large or small. Finally, the third and most descriptive, Table VIIIc of Cluster Profiles (Means), provides a detailed characterization of each group. For example, Cluster 1 is distinguished by having the highest average screen time but the lowest physical activity, while other clusters are defined by different combinations of age, sleep, and activity, allowing for the identification of unique behavioral profiles among individuals in the dataset.

B. Comparison of Methodologies

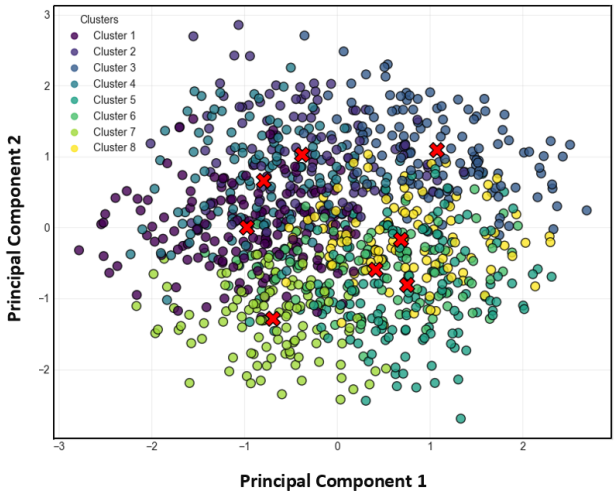
In this regard, Table IX shows the main differences between the most widely used data mining methodologies: KDD, SEMMA, and CRISP-DM. While KDD focuses on knowledge discovery from large volumes of data without a strict methodological structure, SEMMA offers a practical approach but is limited to the use of proprietary tools such as SAS and strictly technical phases. In contrast, CRISP-DM stands out for its comprehensive, iterative, and flexible nature,



(a) Silhouette score evaluation.



(b) Elbow method (Inertia).



(c) Cluster visualization (k=8).

Fig. 5. Evaluation and visualization of k-means clusters.

TABLE VIII. CLUSTER ANALYSIS RESULTS

(A) METRICS PER k

k	Inertia	Silhouette
2	3245.03	0.18
3	2800.44	0.17
4	2416.18	0.19
5	2139.13	0.19
6	1899.30	0.20
7	1746.23	0.21
8	1563.09	0.22
9	1451.26	0.22
10	1363.39	0.21

(B) DISTRIBUTION

Cluster	Count	Percentage (%)
0	155	15.50
1	112	11.20
2	133	13.30
3	125	12.50
4	149	14.90
5	105	10.50
6	114	11.40
7	107	10.70

(C) PROFILES (AVERAGES)

Cluster	Age	Screen	Dream	Activity
0	24.06	4.38	6.92	4.82
1	19.87	8.83	8.04	2.12
2	20.71	9.83	7.61	7.83
3	23.86	8.88	5.35	2.95
4	20.03	6.30	4.88	8.20
5	17.59	4.32	7.89	6.38
6	17.80	3.86	5.72	2.45
7	16.72	9.28	5.51	4.22

covering everything from business understanding to model implementation, making it a robust and widely validated standard in the industry. For these reasons, CRISP-DM is selected as the most appropriate methodology for this project, as it guarantees complete and systematic coverage of the data mining process.

V. Discussion

The study described by [14] demonstrated that unsupervised clustering algorithms make it possible to segment the mental health of university students into clearly differentiated profiles, particularly when employing $k = 3$, supported by the silhouette index. However, unlike that work, which integrated classification and clustering into a hybrid model to optimize the prediction of academic performance and psychological well-being, the present research focuses on a purely unsupervised exploratory approach. This approach, by dispensing with prior labels, facilitates the identification of hidden patterns and enables the comparison of how adjusting the number of clusters ($k = 3$ versus $k = 8$) impacts the balance between clarity and level of detail in segmentation. In doing so, it highlights the usefulness of pure clustering for generating initial analytical foundations that could complement hybrid models in future studies [15].

Complementarily, the work of [16] highlights the positive impact of structured interventions on improving psychological well-being, employing a Bayesian model and cluster sampling to evaluate dimensions such as mental health literacy, self-efficacy, and willingness to seek help. The findings of the present study align with these results by showing that segmentation strategies promote the identification of vulnerable groups and, consequently, the strengthening of such dimensions, albeit with nuances related to the academic and social context of the students analyzed. In this sense, it is evident that, regardless of the methodology applied, intervention programs generate consistent effects in promoting student well-being.

On the other hand, the research of [17] introduces an approach based on wearable devices and physiological signals (EEG, EDA, HR, and SKT), achieving accuracy levels above 99% in stress classification in controlled environments using the MIST. While these results demonstrate the effectiveness of real-time monitoring, they present limitations regarding their applicability in everyday scenarios. In contrast to this approach, the results obtained here prioritize contextual variables such as lifestyle and academic habits, which, although yielding lower levels of accuracy, provide a broader and more representative perspective of the students' real environment. Thus, a contrast is established between the high accuracy of wearables under experimental conditions and the practical applicability of contextual factors in real educational settings.

Likewise, compared with studies that employed questionnaires and classical classification algorithms such as Naïve Bayes, Decision Trees, Random Forest, and SVM, achieving accuracies between 71% and 78.9% for predicting anxiety levels [18] [19], the results obtained in the present study provide an alternative approach by integrating clustering techniques and contextual analysis. Although supervised algorithms allow direct predictions of specific symptoms, clustering offers a complementary perspective by identifying group patterns that transcend individual characteristics, thus providing a more solid foundation for preventive interventions of a collective nature.

In the same vein, studies such as those of [20] [21] employed supervised models including multiple linear regression, random forest, and multilayer neural networks to predict depression, suicidal ideation, and stress levels, highlighting specific factors such as financial concerns or somatic symptoms as key predictors. In contrast, the present study emphasizes group analysis through clustering, revealing how students can be organized into profiles of shared well-being and risk. This perspective makes it possible to move beyond the prediction of individual cases and focus instead on the understanding of collective trends that can guide broader preventive interventions tailored to different segments of the student population.

When contrasting the findings of Rois [22] and Saha et al. [23] with the results of this study, a convergence emerges regarding the relevance of external factors in explaining student mental well-being. While those authors identified physiological variables such as heart rate, blood pressure, or sleep quality using supervised algorithms—highlighting random forest with an accuracy of 89.72% and an AUC of 0.8715—as well as the utility of social media for anticipating mental health consultations with a correlation of $r = 0.86$, this work applied unsupervised clustering techniques that allowed students to be segmented into groups with similar risk and mental prosperity characteristics. Unlike predictive approaches based on supervised models or large-scale data analysis, the results presented here offer an exploratory perspective that reveals collective patterns without requiring prior labels, complementing previous findings by facilitating a comprehensive understanding of how academic, personal, and contextual factors combine into differentiated mental health profiles.

Similarly, the studies by Abdul [24] [25] used physiological variables, lifestyle habits, and large-scale social media data, supported by algorithms such as random forest, SVM, and time series models, achieving remarkable performance metrics (AUC of 0.8715 and correlations up to $r = 0.86$). However, while those approaches privilege individual prediction, the results presented here underscore the capacity of clustering to provide a global view of how students are distributed across different profiles of risk or well-being. Thus, a methodological complementarity is proposed: supervised models are useful for anticipating individual cases, while unsupervised techniques allow for the understanding of collective structures that strengthen the design of prevention strategies and comprehensive monitoring.

Finally, the findings are consistent with those reported by [26] [27], who demonstrated the usefulness of clustering in segmenting university mental health profiles based on factors such as alcohol consumption, perceived stress, or academic cohorts. While these authors applied k-means and other clustering methods with specific variables, the present study jointly integrates academic and personal factors, expanding the scope of analysis and providing a more holistic perspective of student well-being. Taken together, the results reaffirm that

TABLE IX. COMPARISON OF DATA MINING METHODOLOGIES

Criterion	KDD	SEMMA	CRISP-DM
Main Focus	Knowledge discovery from large volumes of data	Analytical process focused on statistical modeling	Standardized data mining process applied to different domains
Methodological Structure	Does not have rigidly defined phases, more conceptual	Technical phases: Sample, Explore, Modify, Model, Assess	Clear phases: Business Understanding, Data Understanding, Preparation, Modeling, Evaluation, Deployment
Flexibility	Low flexibility; requires adaptation to context	Limited to projects using SAS software	High flexibility and iterativity, applicable to diverse contexts
Scope	More theoretical discovery than practical implementation	Focused on exploratory analysis and model building	Covers the entire lifecycle: from business to implementation
Advantages	Pioneer in formalizing the process of pattern discovery	Direct and practical in SAS projects	Industry standard, robust, documented, and widely used
Limitations	Lacks standardization for real projects	Dependence on proprietary software, not very adaptable to other environments	May be more extensive and complex compared to simpler methodologies

clustering is not only effective for classification but also for generating segmentations that enable the design of timely and tailored interventions for differentiated profiles within the university population.

VI. CONCLUSION

Mental health in the university setting has shown various issues that significantly affect the academic and personal lives of students. These difficulties not only influence academic performance but also impact behavior, interpersonal relationships, and emotional well-being. In this context, the research conducted proposed the creation of a predictive model based on machine learning, using the k-means algorithm as the main approach. To develop the proposed model, a dataset in CSV format, oriented toward university students' mental health, was extracted from the Kaggle platform. The research was structured following the CRISP-DM methodology, which establishes five stages in its development. The first stage refers to business understanding based on the identification of the problem. As a second stage, an analysis of the most relevant variables from the dataset was conducted, applying criteria such as mean, maximum, minimum, among others. The third stage involved analyzing outliers, performing data cleaning, and transforming categorical variables into numerical variables for further evaluation. In the fourth stage, the k-means algorithm was applied to cluster the data and evaluate the optimal number of groups created. Finally, the fifth stage provides an evaluation of metrics based on the established clusters.

The results obtained confirm that cluster analysis through PCA, along with metrics such as Silhouette and inertia, makes it possible to identify significant patterns within the student population. It is observed that configurations such as $k = 3$ and $k = 8$ generate consistent groupings, differentiated by key variables such as screen time, hours of sleep, and level of physical activity. These segmentations not only facilitate the understanding of students' distribution and behavior in the academic environment but also help identify subgroups with specific needs or habits. Altogether, this approach provides relevant information for designing academic support strategies and programs aimed at well-being and mental health, contributing to a deeper understanding of the factors affecting university performance and adaptation.

Based on these findings, it is recommended that future research incorporate additional variables that allow for a more comprehensive characterization of students, such as eating habits, use of social networks, socioeconomic level, and resilience. Likewise, it is advisable to complement cluster analysis with different algorithms such as DBSCAN, Agglomerative Clustering, or Gaussian Mixture Models, as well as to explore advanced dimensionality reduction techniques such as t-SNE or UMAP to improve data visualization. It is also suggested to evaluate cluster stability through cross-validation or bootstrap techniques, integrate temporal data to analyze changes over time, and combine clustering with supervised predictive models to anticipate academic or mental health risks. Finally, complementing the study with interactive tools or dashboards would facilitate the interpretation and application of results, strengthening the model's usefulness for decision-making in educational environments.

The developed model presents certain limitations that must be considered when interpreting the results. First, it is based solely on the variables available in the original dataset; therefore, relevant external factors for mental health and academic performance may not be reflected. Moreover, since this is an unsupervised cluster analysis, the assignment of students to groups depends on the algorithm configuration and the choice of the number of clusters, which may generate some variability in the results. Another limitation is that the model does not incorporate temporal data, so it does not allow for the evaluation of how students' behavioral patterns change over time. Lastly, although quality metrics such as Silhouette and inertia help evaluate cluster cohesion and separation, they do not guarantee an absolute interpretation of the practical relevance of each group, making it necessary to complement the analysis with input from education and mental health experts.

ACKNOWLEDGMENT

Thanks to the University of Sciences and Humanities, which made this project a reality.

REFERENCES

[1] K. Wantala, F. C. L. S. Go, V. C. C. Garcia, P. Chirawatkul, N. Chanlek, P. Kidkhunthod, R. R. M. Abarca, and M. D. G.

de Luna, "Low thermal oxidation of gaseous toluene over cu/ce single-doped and co-doped oms-2 on different synthetic routes," *Chemical Engineering Communications*, vol. 211, pp. 350–365, 2024, doi:10.1080/00986445.2022.2050710.

[2] D. A. Doubblestein, B. A. Spinelli, A. Goldberg, C. A. Larson, and A. M. Yorke, "Use of outcome measures by certified lymphedema therapists with survivors of breast cancer with breast cancer-related lymphedema," *Rehabilitation Oncology*, vol. 41, pp. 34–46, 1 2023, doi:10.1097/01.REO.0000000000000310.

[3] E. K. Ponce, M. F. Cruz, and L. Andrade-Arenas, "Machine learning applied to prevention and mental health care in peru," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 1, pp. 823 – 831, 2022, doi:10.14569/IJACSA.2022.0130196.

[4] C. Zheng, S. Liu, L. Song, Z. Xu, J. Guo, Y. Ma, Q. Ju, and J. Wang, "Comparison of sensible and latent heat fluxes from optical-microwave scintillometers and eddy covariance systems with respect to surface energy balance closure," *Agricultural and Forest Meteorology*, vol. 331, 3 2023, doi:10.1016/J.AGRFORMET.2023.109345.

[5] J. A. Harris, P. K. Guntaka, C. J. Niedziela, S. R. Aziz, and S. Afshar, "Interest in global surgery rotations among oral and maxillofacial surgical residents in the united states," *Journal of Dental Education*, vol. 88, pp. 30–41, 1 2024, doi:10.1002/JDD.13394.

[6] C. O. M. Arroyave, D. C. Arango, D. A. Restrepo-Ochoa, and A. C. Calvo, "Positive mental health: between well-being and the development of capabilities," *CES Psicología*, vol. 15, no. 2, 2022, doi:10.21615/cesp.5275.

[7] J. Soihet and A. D. Silva, "Psychological and metabolic effects of food restriction in binge eating disorder," *Nutrição Brasil*, vol. 18, no. 1, 2019, doi:10.33233/nb.v18i1.2563.

[8] M. S. Suzigan, L. da Ressurreição Santos, S. A. Rosa, E. R. Caldas, B. M. D. Araújo, F. de Castro Caixeta, A. A. D. Silva, and M. V. Ferreira, "Neurobiology of anxiety disorders," *Brazilian Journal of Health Review*, vol. 7, no. 1, 2024, doi:10.34119/bjhrv7n1-492.

[9] S. S. Valente, A. V. Padoin, D. S. Valente, C. L. de Sousa Brito, C. C. Mottin, and L. B. Micheletto, "Impact of psychological factors on bariatric surgery failure," *Psico*, vol. 54, no. 1, 2023, doi:10.15448/1980-8623.22.1.39907.

[10] C. Prado, M. Santero, D. Caruso, F. Ortiz, M. S. Zamorano, and V. Irazola, "What are the knowledge gaps and research priorities in mental health of older adults? a mixed-method study using the combined strategies matrix for argentina (meca)," *Global Health Promotion*, vol. 30, pp. 87–94, Mar. 2023, doi:10.1177/17579759221086282.

[11] N. O. Hidayati, E. Widianti, D. A. Amira, Alfiatullatifah, R. H. Pratama, and R. R. N. Asifa, "Elderly in prison: A scoping review of mental health problems," *Enfermeria Global*, vol. 23, pp. 503–513, 2024, doi:10.6018/EGLOBAL.563741.

[12] M. Y. de Fatima Cucho Hidalgo and D. L. J. G. Lola, "Operational management of mental health services for the development of psychological well-being in post-pandemic university students," *Ciencia Latina Revista Científica Multidisciplinar*, vol. 7, pp. 1284–1301, 1 2023, doi:10.37811/CL_RCM.V7I1.4481.

[13] A. Gonzalez-Mora and M. Gomez-Vargas, "Social relationships and mental health of migrant mothers and their children in medellin, colombia," *Revista Virtual Universidad Catolica del Norte*, pp. 140–167, Jan. 2023, doi:10.35575/RVUCN.N68A7.

[14] H. Singh and B. Kaur, "Predicting student performance and its impact on mental health using machine learning," *SSRN Electronic Journal*, 2023, doi:10.2139/ssrn.4356166.

[15] J. Gera, E. B. Marwaha, R. Thareja, R. Thareja, S. Gupta, and A. Jain, "Analysing student's academic performance in relation to psychosocial aspects using ai," *International Journal of Engineering Trends and Technology*, vol. 72, no. 1, 2024, doi:10.14445/22315381/IJETT-V72I1P124.

[16] X. Li, "Short-term rainfall monitoring in mountainous area based on bayesian model and mental health intervention of college students," *Arabian Journal of Geosciences*, vol. 14, no. 16, 2021, doi:10.1007/s12517-021-08010-5.

[17] V. Chandra and D. Sethia, "Machine learning-based stress classification system using wearable sensor devices," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 337–347, 2024, doi:10.11591/ijai.v13.i1.pp337-347.

[18] F. Fang, W. Chung, C. Ventre, M. Basios, L. Kanthan, L. Li, and F. Wu, "Ascertaining price formation in cryptocurrency markets with machine learning," *European Journal of Finance*, vol. 30, no. 1, p. 78–100, 2024, doi:10.1080/1351847X.2021.1908390.

[19] S. S. Rajkishan, A. J. Meitei, and A. Singh, "Role of ai/ml in the study of mental health problems of the students: a bibliometric study," *International Journal of System Assurance Engineering and Management*, vol. 15, no. 5, pp. 5639–5654, 2024, doi:10.1007/s13198-023-02052-6.

[20] N. Meda, S. Pardini, P. Rigobello, F. Visioli, and C. Novara, "Frequency and machine learning predictors of severe depressive symptoms and suicidal ideation among university students," *Epidemiology and Psychiatric Sciences*, vol. 32, p. e55, 2023, doi:10.1017/S2045796023000550.

[21] I. J. Ratul, M. M. Nishat, F. Faisal, S. Sultana, A. Ahmed, and M. A. A. Mamun, "Analyzing perceived psychological and social stress of university students: A machine learning approach," *Heliyon*, vol. 9, no. 6, p. e17307, 2023, doi:10.1016/j.heliyon.2023.e17307.

[22] R. Rois, M. Ray, A. Rahman, and S. K. Roy, "Prevalence and predicting factors of perceived stress among bangladeshi university students using machine learning algorithms," *Journal of Health, Population and Nutrition*, vol. 40, no. 1, p. 50, 2021, doi:10.1186/s41043-021-00276-5.

[23] K. Saha, A. Yousuf, R. L. Boyd, J. W. Pennebaker, and M. D. Choudhury, "Social media discussions predict mental health consultations on college campuses," *Scientific Reports*, vol. 12, no. 1, p. 17928, 2022, doi:10.1038/s41598-021-03423-4.

[24] H. A. Rahman, M. Kwicklis, M. Ottom, A. Amornsriwatanakul, K. H. Abdul-Mumin, M. Rosenberg, and I. D. Dinov, "Machine learning-based prediction of mental well-being using health behavior data from university students," *Bioengineering*, vol. 10, no. 5, p. 575, 2023, doi:10.3390/bioengineering10050575.

[25] Y. Saxena, A. K. Mishra, D. Arora, and R. Devi, "Emotion based mental health classifier for ncr based engineering students," in *Proceedings of the 6th International Conference on Contemporary Computing and Informatics (IC3I 2023)*. IEEE, 2023, pp. 285–290, doi:10.1109/IC3I59117.2023.10397910.

[26] S. Lee, J. Lim, S. Lee, Y. Heo, and D. Jung, "Group-tailored feedback on online mental health screening for university students: using cluster analysis," *BMC Primary Care*, vol. 23, no. 1, p. 33, 2022, doi:10.1186/s12875-021-01622-6.

[27] B. F. Wee, S. Sivakumar, K. H. Lim, W. K. Wong, and F. H. Juwono, "Diabetes detection based on machine learning and deep learning approaches," *Multimedia Tools and Applications*, vol. 83, pp. 24 153–24 185, 2024, doi:10.1007/s11042-023-16407-5.

[28] S. Z. E. Mestari, G. Lenzini, and H. Demirci, "Preserving data privacy in machine learning systems," *Computers and Security*, vol. 137, 2024, doi:10.1016/j.cose.2023.103605.

[29] D. Peral-García, J. Cruz-Benito, and F. J. García-Peñalvo, "Systematic literature review: Quantum machine learning and its applications," *Computer Science Review*, vol. 51, 2024, doi:10.1016/j.cosrev.2024.100619.

[30] J. J. E. Zúñiga, "Application of crisp-dm methodology for geographic segmentation of a public database," *Ingeniería Investigación y Tecnología*, vol. 21, no. 1, pp. 1–15, 2020, doi:10.22201/fi.25940732e.2020.21n1.008.

[31] R. Clancy, D. O'Sullivan, and K. Bruton, "Data-driven quality improvement approach to reducing waste in manufacturing," *TQM Journal*, vol. 35, no. 1, pp. 51–72, 2023, doi:10.1108/TQM-02-2021-0061.

[32] C. E. D. Vanegas, J. C. G. Mejía, F. A. V. Agudelo, and D. E. S. Duran, "A representation based on essence for the crisp-dm methodology," *Computacion y Sistemas*, vol. 27, no. 3, pp. 675–689, 2023, doi:10.13053/CYS-27-3-3446.

[33] J. Brzozowska, J. Pizoń, G. Baytikenova, A. Gola, A. Zakimova, and K. Piotrowska, "Data engineering in crisp-dm process production data – case study," *Applied Computer Science*, vol. 19, no. 3, pp. 83–95, 2023, doi:10.35784/ACS-2023-26.

[34] J. Bokrantz, M. Subramaniyan, and A. Skoogh, "Realising the promises of artificial intelligence in manufacturing by enhancing crisp-dm," *Production Planning and Control*, vol. 35, pp. 2234–2254, 2024, doi:10.1080/09537287.2023.2234882.

[35] V. Krishnaswamy, N. Singh, M. Sharma, N. Verma, and A. Verma, "Application of crisp-dm methodology for managing human-wildlife conflicts: An empirical case study in india," *Journal of Environmental Planning and Management*, vol. 66, no. 11, pp. 2247–2273, 2023, doi:10.1080/09640568.2022.2070460.

[36] A. Pambudi, "Application of crisp-dm using mlr k-fold on stock data of pt. telkom indonesia (persero) tbk: A case study on the indonesia stock exchange 2015–2022," *Jurnal Data Mining dan Sistem Informasi*, vol. 4, no. 1, pp. 1–15, 3 2023, doi:10.33365/JDMSI.V4I1.2462.

[37] V. Plotnikova, M. Dumas, and F. P. Milani, "Applying the crisp-dm data mining process in the financial services industry: Elicitation of adaptation requirements," *Data and Knowledge Engineering*, vol. 139, p. 102013, 5 2022, doi:10.1016/J.DATAK.2022.102013.

[38] A. Cheng, "Evaluating fintech industry's risks: A preliminary analysis based on crisp-dm framework," *Finance Research Letters*, vol. 55, p. 103966, 7 2023, doi:10.1016/J.FRL.2023.103966.

TomDetLeaf: A Realistic Multi-Source Dataset for Real-Time Tomato Leaf Detection

Yassmine Ben Dhiab[1], Mohamed Ould-Elhassen Aoueileyine[2], Abdallah Namoun[3]*, Ridha Bouallegue[4]
Innov'COM Laboratory-Higher School of Communication of Tunis (SUP'COM), University of Carthage, Tunis, Tunisia
AI Center-Faculty of Computer and Information Systems, Islamic University of Madinah, Madinah 42351, Saudi Arabia
Ibn Khaldoun University (UIK), Soukra, Ariana 2036, Tunisia

***Abstract*—Plant diseases remain a major threat to crop productivity, especially where timely diagnosis is difficult. This paper introduces TomDetLeaf, a new annotated dataset designed for tomato leaf detection in diverse agricultural environments, supporting the development of generalizable deep learning models for edge AI deployment. Unlike existing datasets such as PlantVillage, which consist mainly of single-leaf images captured under controlled conditions, TomDetLeaf integrates heterogeneous sources including the Taiwan dataset, climate-controlled greenhouses, hydroponic systems and farm environments. The dataset combines single-leaf and multi-leaf images, realistic backgrounds and varying illumination, addressing a key gap that limits the real-world robustness of current models. To demonstrate its utility, we trained and evaluated YOLOv8 on both the original Taiwan dataset and our proposed TomDetLeaf. Results show that YOLOv8 trained on TomDetLeaf achieved 88.3% mAP@0.5, 81.8% precision, and 82.7% recall, exceeding the Taiwan-subset baseline of 77.4% mAP@0.5, 81.6% precision, and 67.6% recall. This validates the contribution of TomDetLeaf in improving detection accuracy and generalization under realistic conditions. By providing a diverse, deployment-ready dataset, this work bridges the gap between theoretical benchmarks and practical real-time applications.**

***Keywords*—Tomato leaf detection; smart agriculture; dataset; tomato leaf dataset; real-time inference; Edge AI; object detection**

I. Introduction

Ensuring tomato crop health is vital for maintaining sustainable agricultural production and global food security [1]. Tomato plants are highly susceptible to multiple foliar diseases that cause severe yield losses and economic constraints for farmers worldwide [2], [3]. Early and accurate detection of diseased leaves is therefore critical for timely intervention and disease management. Conventional detection methods, such as expert visual inspections or laboratory analysis, are limited by subjectivity, high costs and the lack of accessibility in rural or resource-constrained regions [4].

In recent years, the combination of deep learning (DL) and computer vision has shown promising results for plant disease recognition. Convolutional neural networks (CNNs), trained on large-scale datasets such as PlantVillage, have achieved remarkable classification accuracy, often exceeding 99% [5]. However, these models typically rely on synthetic or laboratory-controlled images with plain backgrounds and uniform lighting, conditions that do not reflect the complexity of real-world agricultural environments. As a result, models trained on such data may perform well in academic benchmarks but fail to generalize when deployed in greenhouses, hydroponic systems or open-field farms [6].

To address these limitations, there is a pressing need for annotated datasets that capture the diversity of real-world conditions, including single and multiple leaves per image, heterogeneous backgrounds, variable lighting and naturally occurring noise [7]. Without such datasets, AI systems risk providing artificially inflated results during testing while underperforming in practical deployment scenarios.

Despite the success of existing datasets such as PlantVillage [8], their controlled acquisition conditions, uniform backgrounds, isolated leaves and synthetic lighting often lead to inflated accuracy that does not generalize to real-world deployment. Models trained on such data frequently fail when tested in farms, greenhouses, or hydroponic environments where multiple leaves overlap, backgrounds vary and environmental noise is present [9]. This critical gap between laboratory benchmarks and field performance motivates the development of TomDetLeaf, a dataset explicitly designed to capture real-world complexity and support the creation of robust detection models for edge deployment [10].

This work introduces TomDetLeaf, a curated dataset designed to bridge this gap by enabling robust tomato leaf detection in diverse contexts. Unlike existing resources, TomDetLeaf merges images from multiple sources, including the Taiwan dataset, a climate-controlled greenhouse and hydroponic environments and it is complemented with distractor images to enhance model generalization [11]. Each tomato leaf in the dataset is annotated for detection tasks, providing a realistic foundation for developing edge-ready models. To demonstrate its effectiveness, we trained YOLOv8 on TomDetLeaf and compared its performance with the same model trained on the original Taiwan dataset [12]. Results show that TomDetLeaf enables a substantial improvement in both mean Average Precision (mAP) and recall, while achieving stronger robustness in real-world scenarios.

Beyond dataset development, this work sets the foundation for an end-to-end pipeline combining detection and classification. In future work, the detected tomato leaves will be cropped and passed to a lightweight classification model capable of distinguishing between healthy and multiple disease categories in real time. The pipeline will be designed for integration into autonomous platforms, such as agricultural robots or UAVs, enabling continuous monitoring of crops directly in the field [13], [14]. This vision positions TomDetLeaf not only as a

*Corresponding authors.

dataset, but as a key enabler of embedded AI solutions for smart agriculture.

The main contributions of this paper are summarized as follows:

- Development of TomDetLeaf, a new annotated dataset for tomato leaf detection, integrating images from controlled and real-world environments with diverse backgrounds, lighting and leaf arrangements.
- Cleaning and curation of existing data sources, including the Taiwan dataset, by removing low-quality and duplicate samples to enhance reliability.
- Incorporation of greenhouse, hydroponic and distractor images, ensuring robustness against non-ideal and heterogeneous agricultural conditions.
- Comprehensive evaluation with YOLOv8, showing significant performance improvements over baseline datasets, confirming the dataset's value for real-time deployment in edge-based agricultural applications.

By filling the dataset gap and validating its utility through rigorous experiments, TomDetLeaf lays the groundwork for reliable end-to-end systems where detection of tomato leaves can serve as the first step toward subsequent disease classification and full integration into autonomous agricultural platforms.

To situate the dataset within a complete decision workflow, Fig. 4 presents the conceptual architecture of the intended system. It comprises image acquisition, tomato-leaf detection with YOLOv8, cropping of detected leaves, a lightweight disease classification module and a final decision layer that reports health status and disease type. The present study empirically evaluates the detection component; the classification component is outlined as future work.

II. Proposed Dataset: TomDetLeaf

The TomDetLeaf dataset was designed to provide a realistic and diverse benchmark for tomato leaf detection, with the specific objective of improving the generalization of deep learning models in real-world agricultural contexts. Unlike existing datasets, which are often captured under controlled laboratory conditions with plain backgrounds, TomDetLeaf integrates heterogeneous sources and environmental conditions [15], [16]. This deliberate design ensures that the dataset reflects the variability of actual agricultural fields, where lighting, leaf density and background noise are far less predictable. The construction process follows a structured pipeline that includes data selection, cleaning, integration of new sources, annotation and the addition of distractor images [17].

A. Taiwan Subset and Cleaning

The first source of TomDetLeaf is the publicly available Taiwan tomato dataset [18], which contains both single-leaf and multi-leaf images. From this dataset, only a relevant subset was retained. The raw images were manually cleaned to remove duplicates, low-quality samples and blurred images that could negatively affect model learning. This cleaning process ensured that the retained data maintained reliability and quality [19]. The Taiwan subset serves as the foundation of TomDetLeaf, contributing a balanced mixture of isolated and overlapping tomato leaves under varying capture conditions.

B. Greenhouse Captures

To complement the Taiwan subset, we collected new high-resolution images from a climate-controlled greenhouse. These images introduce diversity in terms of background textures, controlled yet variable lighting and different plant growth stages. Including greenhouse data ensures that the dataset better reflects the conditions of commercial tomato cultivation systems, where disease detection systems are increasingly deployed. The controlled acquisition process in this environment also provided consistent labeling opportunities, while maintaining realistic challenges such as shadows, leaf overlap and reflections [20].

C. Hydroponic Captures

Another significant contribution to TomDetLeaf comes from hydroponic environments, where tomato plants are grown in nutrient solutions without soil. The images captured in this context differ substantially from those in both controlled greenhouse and field scenarios. Hydroponic systems often include artificial lighting, reflective surfaces and pipes or structural elements in the background, which add to the complexity of the detection task [20]. Incorporating these samples enhances the dataset's heterogeneity and improves the likelihood that trained models will generalize to a broad spectrum of agricultural settings [21].

D. Distractor Images

In addition to tomato-specific samples, TomDetLeaf includes a curated set of distractor images consisting of non-tomato leaves, stems and other irrelevant agricultural objects. These negative examples are essential to reduce model bias and prevent false positives during deployment. In real-world conditions, object detection systems are frequently exposed to cluttered backgrounds containing various types of vegetation and objects unrelated to tomato crops. By incorporating distractor images into the training process, TomDetLeaf strengthens the ability of models to distinguish tomato leaves from irrelevant elements, thereby improving robustness in practical applications.

E. Annotation Process

All images were consistently annotated using the Roboflow platform. Bounding boxes were applied to each tomato leaf with a single unified label, Tomato-leaf. The decision to annotate only one class simplifies the dataset for the detection stage, which we consider the foundation of an end-to-end pipeline. This modular approach allows subsequent classification models to focus exclusively on cropped leaf regions, where disease type or health status can later be determined. The consistency of annotations across heterogeneous sources ensures clarity and usability for researchers and developers aiming to deploy detection systems on embedded devices.

F. Dataset Overview

Fig. 1 presents the complete construction pipeline of TomDetLeaf, while Fig. 2 shows representative examples of images drawn from both the Taiwan dataset and the proposed TomDetLeaf dataset. The latter clearly demonstrates its

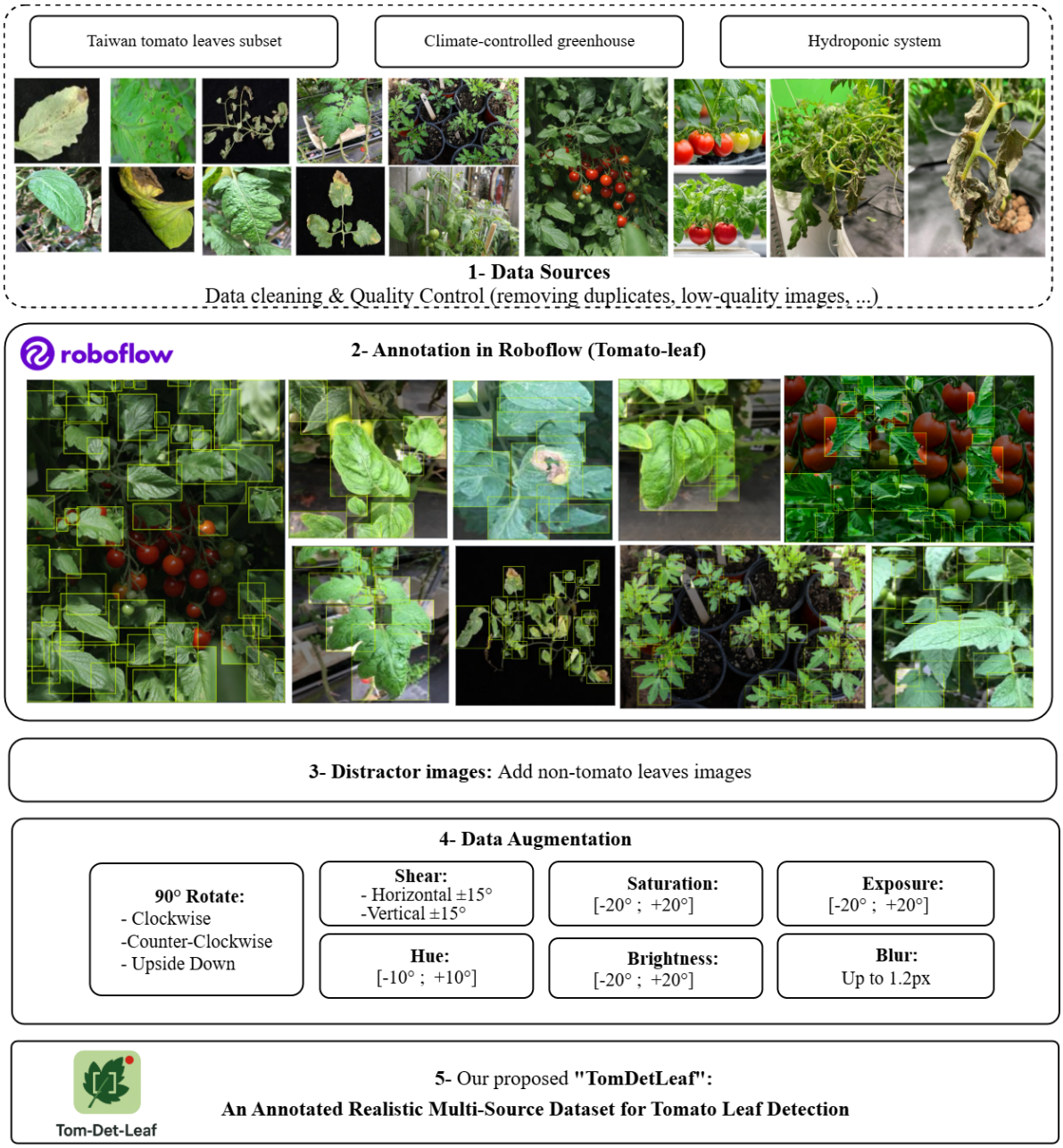


Fig. 1. Construction pipeline of the proposed TomDetLeaf dataset. The dataset integrates cleaned subsets of the Taiwan dataset with new captures from climate-controlled greenhouses and hydroponic systems, complemented by distractor images and consistently annotated for tomato leaf detection.

enhanced variability, combining single and multiple leaves, natural and artificial backgrounds and diverse environmental conditions. This heterogeneity makes TomDetLeaf a unique and valuable resource for training and evaluating models intended for TinyML and Edge AI deployments in agriculture [22], [23].

G. Dataset Statistics

Table I summarizes the composition of TomDetLeaf after cleaning. Counts reflect the final pool used prior to splitting.

The present study addresses leaf detection only, i.e. all leaves share a single Tomato-leaf class. The healthy/diseased counts in Table I were not used during training or evaluation; they are reported to show coverage across environments and

TABLE I. TOMDETLEAF COMPOSITION BY SOURCE AFTER CLEANING. H: HEALTHY, D: DISEASED

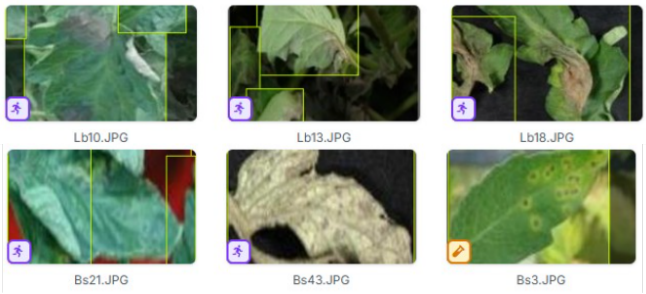
Source	Images	H	D
Taiwan subset (cleaned)	417	63	354
Greenhouse captures	182	78	104
Hydroponic captures	127	74	53
Distractor scenes (negatives)	34	–	–
Total	**760**	**215**	**511**

to motivate the forthcoming classification release.

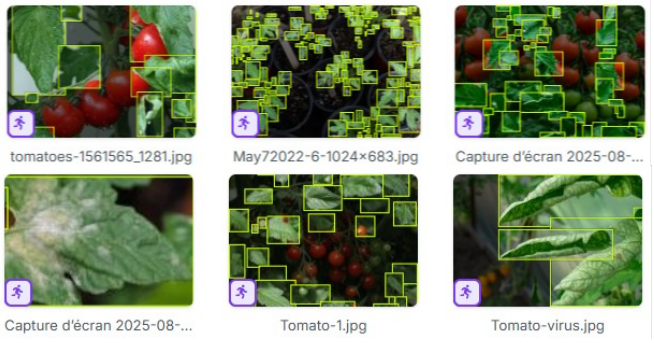
To indicate scene complexity at a glance, TomDetLeaf (non-augmented) comprises approximately one third single-leaf images and two thirds multi-leaf images (see Table II).

TABLE II. SINGLE- VS. MULTI-LEAF COMPOSITION OF TOMDETLEAF (NON-AUGMENTED)

Scene type	Share	Images (of 760)
Single-leaf	≈33%	≈250
Multi-leaf	≈67%	≈510



(a) Samples from the Taiwan dataset



(b) Samples from the proposed TomDetLeaf dataset

Fig. 2. Examples of tomato leaf images: (a) Samples from the Taiwan dataset, (b) Samples from the proposed TomDetLeaf dataset. The TomDetLeaf dataset integrates single and multiple leaves, diverse backgrounds and images from real-world environments such as farms, climate-controlled greenhouses and hydroponics.

PlantVillage (tomato subset) [8] is a large classification corpus of single-leaf images captured on largely uniform backgrounds with image-level labels only (no bounding boxes). By design it is not detection-ready and would require re-annotation to train a detector. In contrast, TomDetLeaf provides bounding boxes for a single Tomato-leaf class, mixes single- and multi-leaf scenes (approximately one third vs. two thirds), and includes complex, real-world backgrounds across greenhouse and hydroponic settings. Consequently, direct mAP comparison to PlantVillage is not meaningful; it serves instead as a scale reference for classification. Key differences are summarized in Table III.

TABLE III. TOMDETLEAF VS. PLANTVILLAGE (TOMATO SUBSET)

Aspect	TomDetLeaf	PlantVillage–Tomato
Primary task	Detection	Classification
Annotation	Boxes (1 class)	Image labels
Leaf multiplicity	Single+Multi (≈1/3:≈2/3)	Single only
Backgrounds	Complex, real-world	Plain, uniform
Environments	Greenhouse, hydroponic, farm	Controlled
Images (raw)	760	18,164
Detection-ready	Yes	No (re-annotation)

III. EXPERIMENTAL SETUP

To evaluate the effectiveness of the proposed TomDetLeaf dataset, we conducted experiments using YOLOv8, a state-of-the-art object detection model widely adopted for lightweight and real-time applications. YOLOv8 was selected due to its balance between accuracy and computational efficiency, making it suitable for assessing detection performance in agricultural edge scenarios. The implementation was based on the Ultralytics YOLOv8 framework with PyTorch backend.

A. Datasets for Comparison

Two datasets were considered in this study:

1) Taiwan dataset: A publicly available dataset of tomato plants [18], annotated to include a single bounding-box class labeled Tomato-leaf. Only the relevant subset was retained, with noisy, poor-quality and duplicate images removed.

2) TomDetLeaf dataset (Proposed): A newly curated dataset combining selected and cleaned images from the Taiwan dataset with additional images collected from climate-controlled greenhouses and hydroponic environments. The dataset captures a wider variety of conditions, including single-leaf and multi-leaf settings, diverse backgrounds and real-world environmental variability. All images were annotated with a single class Tomato-leaf.

For both datasets the detector is health-agnostic (single Tomato-leaf class); the healthy/diseased composition in Table I does not alter the training labels used here.

B. Training Setup

Both datasets were trained under identical conditions to ensure a fair comparison. Images were resized to **225**×**225** and the data were split **70%/20%/10%** into training, validation, and test sets (**532/152/76** images, respectively). Training ran for **100** epochs with a batch size of **64**, using the **Adam** optimizer (initial learning rate **0.001**). Data augmentation was applied to the *training* split only (random horizontal flip, mild geometric transforms and photometric jitter) and performed online at load time, yielding an effective **3**× increase in training samples per epoch; this corresponds to **1,596** effective training images and an overall effective dataset size of **1,824** when counting augmented samples, while validation and test sets remained unchanged.

C. Evaluation Metrics

The performance of the object detection models was evaluated using three standard metrics: Precision, Recall and mean Average Precision (mAP) [24], [25]. These metrics are widely adopted in computer vision benchmarks and provide

complementary insights into model reliability and robustness in real-world scenarios.

Precision measures the ability of the model to correctly identify tomato leaves among all predicted bounding boxes. It is defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

where, TP denotes true positives (correct detections) and FP false positives (incorrect detections). A higher precision indicates that fewer non-leaf objects are misclassified as leaves.

Recall quantifies the proportion of correctly detected tomato leaves among all ground-truth annotations:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

where, FN represents false negatives (missed detections). High recall is particularly critical in agricultural monitoring since missing diseased leaves may prevent timely intervention.

Mean Average Precision (mAP) provides a holistic measure that balances precision and recall. It is based on the area under the Precision–Recall (PR) curve. For a single class, the Average Precision (AP) is computed as:

$$AP = \int_0^1 P(R), \mathrm{d}R \tag{3}$$

where, $P(R)$ is the precision as a function of recall. For multi-class tasks, the mean of AP across all classes yields the mAP. In practice, different intersection-over-union (IoU) thresholds are applied to determine whether a predicted bounding box matches a ground-truth object.

$mAP@50$ corresponds to AP calculated at a fixed IoU threshold of 0.5, following the Pascal VOC standard. It is relatively lenient, considering a detection correct if the overlap with ground truth is at least 50%.

$mAP@90$ applies a stricter threshold of 0.9, requiring near-perfect overlap between predicted and ground-truth bounding boxes.

$mAP@[.5 : .95]$, the COCO standard, averages AP over IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05. This provides a more comprehensive assessment of detection quality across varying levels of localization.

By reporting precision, recall and mAP under multiple IoU thresholds, we provide a complete picture of detection performance, highlighting both the ability to identify leaves accurately (precision) and to avoid missing them (recall), as well as the localization accuracy of bounding boxes (mAP). Unless otherwise stated, reported mAP corresponds to mAP@0.5 (Pascal VOC).

TABLE IV. DETECTION PERFORMANCE OF YOLOV8 TRAINED ON TAIWAN AND TOMDETLEAF DATASETS

Dataset	Precision (%)	Recall (%)	mAP@0.5 (%)
Taiwan dataset	81.6	67.6	77.4
TomDetLeaf (Proposed)	81.8	82.7	88.3

IV. RESULTS

Table IV presents the comparative performance of YOLOv8 [26] trained on the Taiwan dataset and on the proposed TomDetLeaf dataset.

The results demonstrate that TomDetLeaf enables significantly stronger detection performance, with an improvement of +10.9% mAP@0.5 and +15.1% recall compared to the Taiwan dataset. While precision remained similar, recall improved markedly, indicating that models trained on TomDetLeaf capture a larger proportion of true positives under diverse conditions.

An important observation from our experiments is the significant improvement in recall (+15.1%) when training YOLOv8 on TomDetLeaf compared to the Taiwan dataset. In agricultural disease monitoring, recall is particularly critical: missing a diseased leaf can allow pathogens to spread unchecked, whereas a false alarm (lower precision) may only result in an unnecessary inspection. Thus, the higher recall achieved by TomDetLeaf-trained models underscores its value for real-world disease detection systems, where early and reliable identification of all potentially diseased leaves is more important than minimizing false positives.

Qualitative results are illustrated in Fig. 3, showing YOLOv8 predictions on complex real-world tomato plant images. Models trained on TomDetLeaf generalize better to challenging backgrounds, overlapping leaves and mixed plant structures, highlighting the importance of dataset diversity for real-world deployment.

A. Error Analysis and Cross-Dataset Checks

We examined generalization and typical failure modes under the same evaluation protocol as Table IV. A model trained on TomDetLeaf maintains its performance on complex scenes and attains even higher scores on the Taiwan test set, which largely contains simpler, single-leaf images and plain backgrounds. Conversely, a model trained on the Taiwan subset degrades on TomDetLeaf due to background clutter, occlusion and mixed structures.

TABLE V. CROSS-DATASET EVALUATION (MAP@0.5, PRECISION, RECALL)

Training set	Evaluation set	mAP@0.5 (%)	Precision (%)	Recall (%)
TomDetLeaf	TomDetLeaf	88.3	81.8	82.7
TomDetLeaf	Taiwan	92.1	86.4	88.9
Taiwan	Taiwan	77.4	81.6	67.6
Taiwan	TomDetLeaf	65.2	79.8	55.1

Qualitative inspection indicates that false negatives are primarily associated with heavy overlap/occlusion in multi-leaf clusters (approximately 40% of missed instances), small



Fig. 3. YOLOv8 detection results on sample images from the proposed TomDetLeaf dataset. The model demonstrates accurate detection in complex backgrounds, validating the robustness of the dataset.

leaves (height < 32 px; approximately 30%), backlit or low-contrast foliage against bright backgrounds (approximately 20%), and truncation at image borders (approximately 10%). False positives arise mainly from confusion with petioles/stems and tomato fruit (approximately 50%), non-tomato foliage present in aisles or background (approximately 30%) and specular reflections from hydroponic hardware (approximately 20%). We also observed instances where nearby leaves are suppressed during non-maximum suppression in very crowded scenes, which selectively reduces recall in multi-leaf images, the scenario TomDetLeaf was designed to represent.

PlantVillage provides image-level labels only; it is therefore unsuitable for a formal detection benchmark without additional bounding-box annotation. A qualitative probe on single-leaf PlantVillage images suggested that the TomDetLeaf-trained detector localizes the visible leaf reliably, but we do not report mAP for this corpus.

V. Discussion

Fig. 4 summarizes the envisaged end-to-end pipeline for tomato-leaf monitoring. The detector isolates candidate leaves that are subsequently provided to a lightweight classifier. This modular design supports embedded deployment because detection and classification can be optimized independently. The experiments reported here target the detection stage.

In addition, the cross-dataset results in Table V show that models trained on TomDetLeaf transfer reliably to simpler domains (Taiwan), whereas the inverse transfer degrades markedly, underscoring the value of training on realistic, cluttered scenes.

Across all capture settings, both healthy and diseased foliage are present (Table I), which reduces environment–label confounding and supports the observed generalization.

Together with the single- versus multi-leaf composition (Table II), this coverage limits scene-type bias and helps explain the improved recall on complex images.

The experimental results highlight the importance of dataset diversity and realism for developing object detection models that generalize to real-world agricultural environments. Although the Taiwan dataset is commonly used in plant leaf detection tasks, its limited variability in terms of background conditions and image diversity constrained the performance of YOLOv8, particularly in recall (67.6%). This weakness became evident when testing in complex scenarios with multiple overlapping leaves, heterogeneous lighting and natural backgrounds. Models trained on such restricted datasets tend to achieve high accuracy in controlled settings but fail to maintain robustness in deployment contexts.

By contrast, the proposed TomDetLeaf dataset demonstrated a marked improvement in detection performance, raising recall to 82.7% and overall mAP to 88.3%. This represents a relative gain of 10.9 percentage points in mAP and 15.1 percentage points in recall compared to the Taiwan dataset. The improvement in recall is particularly significant, as it indicates that fewer leaves are missed during inference, which is critical in applications such as early disease diagnosis where every missed detection can affect the effectiveness of downstream classification and decision-making.

The strength of TomDetLeaf lies in its deliberate design to reflect realistic agricultural conditions. By merging cleaned

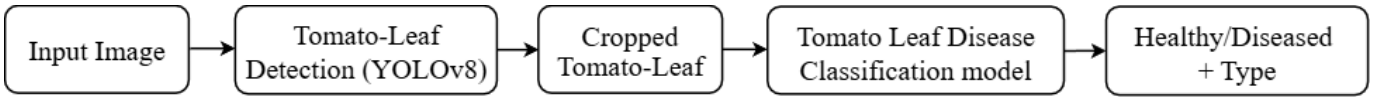


Fig. 4. Conceptual end-to-end pipeline for tomato-leaf monitoring. The system comprises image acquisition, detection with YOLOv8, cropping of detected leaves, a lightweight disease classification module and a final decision reporting health status and disease type. The present work evaluates the detection stage; the classification stage is reserved for future work.

subsets of the Taiwan dataset with additional images collected from climate-controlled greenhouses and hydroponic systems, the dataset introduces multiple layers of complexity: single-leaf and multi-leaf compositions, variations in background and differences in lighting and plant growth environments. Training on such data enables models to encounter challenging scenarios that mimic real-world deployment, thereby enhancing robustness and generalization.

Another important aspect is that TomDetLeaf focuses on leaf-level detection as a first stage of an end-to-end pipeline for plant health monitoring. In practice, robust leaf detection forms the foundation for subsequent disease classification, which can be performed on cropped leaf regions. This approach is well aligned with embedded and TinyML systems, where modular detection–classification pipelines allow real-time inference on resource-constrained edge devices such as robots or UAVs [27], [28]. Therefore, the dataset not only improves detection metrics but also provides a realistic training ground for the next step of integrating lightweight disease classification models [29].

Two practical limitations are worth noting. First, the current release uses a single detection class without disease labels; these labels are planned for the next version. Second, images are resized to 225×225, which can under-represent very small leaves; increasing the input resolution, adopting multi-scale training, or tuning the NMS IoU threshold are straightforward mitigations for future experiments.

Finally, qualitative results confirm that TomDetLeaf-trained models are able to detect tomato leaves under highly cluttered, natural conditions, with consistent bounding-box precision even in the presence of fruits, stems and overlapping vegetation. This demonstrates the practical advantage of training with diverse and realistic data, bridging the gap between controlled research benchmarks and real-world deployment in smart agriculture [30].

VI. Conclusion and Future Work

This work introduced TomDetLeaf, a customized and diverse tomato leaf detection dataset designed to improve the robustness and generalization of object detection models for real-world agricultural applications. By integrating cleaned subsets of existing datasets with newly collected images from climate-controlled greenhouses and hydroponic systems, TomDetLeaf provides a richer representation of realistic field conditions, including single- and multi-leaf scenarios, varied lighting and heterogeneous backgrounds. Experimental evaluation with YOLOv8 demonstrated that models trained on TomDetLeaf outperformed those trained on the Taiwan dataset, achieving an mAP of 88.3% and recall of 82.7%, compared to 77.4% and 67.6%, respectively. These results underline the importance of dataset realism in enabling reliable real-time deployment. Cross-dataset checks further confirmed that TomDetLeaf-trained detectors generalize to simpler domains, while reverse transfer degrades on cluttered scenes.

The dataset is intended as a foundation for modular end-to-end pipelines in smart agriculture, where reliable leaf detection constitutes the first stage of disease monitoring on embedded and edge devices. To facilitate downstream benchmarking, we also report per-source healthy/diseased counts and the single- versus multi-leaf composition to support balanced, reproducible splits. By improving detection accuracy and recall, TomDetLeaf contributes to enabling practical, low-cost solutions for autonomous platforms such as robots and UAVs.

Future work will extend this contribution in several directions. First, we will evaluate the dataset using additional state-of-the-art detection models to benchmark its generalizability across architectures. Second, we are preparing an expanded annotated version of TomDetLeaf that includes disease-specific labels for tomato leaves, allowing simultaneous detection and classification of healthy and diseased samples. Finally, we aim to implement and validate a complete end-to-end pipeline, combining detection and classification on embedded hardware platforms to demonstrate real-time disease diagnosis in realistic agricultural scenarios.

References

[1] Rehman, Abdul, et al. ”Sustainable agricultural practices for food security and ecosystem services.” Environmental Science and Pollution Research 29.56 (2022): 84076-84095.

[2] Shukla, Smriti, et al. ”Challenges faced by farmers in crops production due to fungal pathogens and their effect on Indian economy.” Fungal diversity, ecology and control management. Singapore: Springer Nature Singapore, 2022. 495-505.

[3] Singhal, Pankhuri, et al. ”Mixed infection of plant viruses: diagnostics, interactions and impact on host.” Journal of Plant Diseases and Protection 128.2 (2021): 353-368.

[4] Thangaraj, Rajasekaran, et al. ”Artificial intelligence in tomato leaf disease detection: a comprehensive review and discussion.” Journal of Plant Diseases and Protection 129.3 (2022): 469-488.

[5] Dhaka, Vijaypal Singh, et al. ”A survey of deep convolutional neural networks applied for prediction of plant leaf diseases.” Sensors 21.14 (2021): 4749.

[6] Xu, Mingle, et al. ”Embracing limited and imperfect training datasets: opportunities and challenges in plant disease recognition using deep learning.” Frontiers in Plant Science 14 (2023): 1225409.

[7] Cheng, Hsueh-Hung, et al. ”Identifying tomato leaf diseases under real field conditions using convolutional neural networks and a chatbot.” Computers and Electronics in Agriculture 202 (2022): 107365.

[8] Hughes, David, and Marcel Salathé. ”An open access repository of images on plant health to enable the development of mobile disease diagnostics.” arXiv preprint arXiv:1511.08060 (2015).

[9] Bhargava, Anuja, et al. ”Plant leaf disease detection, classification, and diagnosis using computer vision and artificial intelligence: A review.” IEEE access 12 (2024): 37443-37469.

[10] Akbar, Jalal Uddin Md, et al. "A comprehensive review on deep learning assisted computer vision techniques for smart greenhouse agriculture." IEEE Access 12 (2024): 4485-4522.

[11] Fuentes-Peñailillo, Fernando, et al. "New generation sustainable technologies for soilless vegetable production." Horticulturae 10.1 (2024): 49.

[12] Hussain, Muhammad. "Yolov5, yolov8 and yolov10: The go-to detectors for real-time vision." arXiv preprint arXiv:2407.02988 (2024).

[13] Lytridis, Chris, et al. "An overview of cooperative robotics in agriculture." Agronomy 11.9 (2021): 1818.

[14] Kim, Jeongeun, et al. "Unmanned aerial vehicles in agriculture: A review of perspective of platform, control, and applications." Ieee Access 7 (2019): 105100-105115.

[15] Nobori, Tatsuya, and Kenichi Tsuda. "The plant immune system in heterogeneous environments." Current Opinion in Plant Biology 50 (2019): 58-66.

[16] Thangaraj, Rajasekaran, et al. "Artificial intelligence in tomato leaf disease detection: a comprehensive review and discussion." Journal of Plant Diseases and Protection 129.3 (2022): 469-488.

[17] Zhu, Zheng, et al. "Distractor-aware siamese networks for visual object tracking." Proceedings of the European conference on computer vision (ECCV). 2018.

[18] Huang, Mei-Ling, and Ya-Han Chang. "Dataset of tomato leaves." Mendeley Data 1.10.17632 (2020).

[19] García, Salvador, et al. "Big data preprocessing: methods and prospects." Big data analytics 1.1 (2016): 9.

[20] Malabadi, Ravindra B., et al. "Greenhouse farming: Hydroponic vertical farming-Internet of Things (IOT) Technologies: An updated review." World Journal of Advanced Research and Reviews 23.02 (2024): 2634-2686.

[21] Ragaveena, S., A. Shirly Edward, and U. Surendran. "Smart controlled environment agriculture methods: A holistic review." Reviews in Environmental Science and Bio/Technology 20.4 (2021): 887-913.

[22] Hizem, Moez, et al. "Reliable ECG Anomaly Detection on Edge Devices for Internet of Medical Things Applications." Sensors 25.8 (2025): 2496.

[23] Dhiab, Yassmine Ben, et al. "Edge-based human activity recognition: A novel approach using spectral analysis and deep learning." 2024 International Wireless Communications and Mobile Computing (IWCMC). IEEE, 2024.

[24] Goutte, Cyril, and Eric Gaussier. "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation." European conference on information retrieval. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[25] Yue, Yisong, et al. "A support vector method for optimizing average precision." Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. 2007.

[26] Hussain, Muhammad. "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection." Machines 11.7 (2023): 677.

[27] Elhanashi, Abdussalam, et al. "Advancements in TinyML: Applications, limitations, and impact on IoT devices." Electronics 13.17 (2024): 3562.

[28] Jouini, Oumayma, et al. "A survey of machine learning in edge computing: Techniques, frameworks, applications, issues, and research directions." Technologies 12.6 (2024): 81.

[29] Khan, Zohaib, Yue Shen, and Hui Liu. "ObjectDetection in Agriculture: A Comprehensive Review of Methods, Applications, Challenges, and Future Directions." Agriculture 15.13 (2025): 1351.

[30] Soussi, Abdellatif, et al. "Smart sensors and smart data for precision agriculture: a review." Sensors 24.8 (2024): 2647.

Comparative Analysis of Statistical, Machine Learning, and Deep Learning Approaches for Frost Prediction in the Peruvian Altiplano

Fred Torres-Cruz[1], Dina Maribel Yana-Yucra[2], Richar Andre Vilca-Solorzano[3]
Postgraduate Unit-Faculty of Ingeniería Estadística e Informática,
Universidad Nacional del Altiplano de Puno, Puno, Peru[1]
Professional School-Faculty of Ingeniería Estadística e Informática,
Universidad Nacional del Altiplano de Puno, Puno, Peru[2,3]

***Abstract*—Frost events represent a critical climatic hazard for agricultural systems in the Peruvian highlands, impacting approximately 74% of rural communities in the Puno region. This research addresses the question of whether machine learning (ML) and deep learning (DL) approaches can significantly outperform traditional statistical methods for frost prediction in extreme high-altitude tropical conditions, achieving sufficient accuracy for operational early warning systems. We present a comprehensive evaluation of twelve forecasting models for predicting daily minimum temperatures, utilizing NASA POWER satellite data (2000-2025) from thirteen meteorological stations across the Altiplano plateau (121,056 observations). The study implements and compares traditional statistical approaches (SARIMAX, Holt-Winters, Prophet, STL+ARIMA), machine learning algorithms (Random Forest, Support Vector Machines, XGBoost), deep neural network architectures (Multilayer Perceptron, LSTM, 1D-CNN), a hybrid SARIMA+ANN model, and an optimized ensemble approach. The ensemble model, integrating XGBoost, LSTM, and Random Forest through weighted averaging, demonstrated superior performance with RMSE=1.65°C and TSS=0.87, representing a 35% improvement over the best-performing statistical method. Individual analysis revealed XGBoost achieved RMSE=1.78°C with exceptional feature interaction modeling, while LSTM networks exhibited remarkable temporal pattern recognition with recall=0.88 for frost event detection. These findings validate the effectiveness of nonlinear approaches for operational forecasting under extreme climatic conditions and offer a robust framework for early warning systems that could substantially mitigate agricultural losses in vulnerable high-altitude communities.**

***Keywords*—Frost prediction; machine learning; deep learning; ensemble methods; Altiplano; agricultural early warning systems**

I. Introduction

Frost events, characterized by air temperatures dropping to or below 0°C at standard meteorological height (2 meters) in high-altitude regions exceeding 2,500 meters above sea level [1], constitute a recurring environmental hazard throughout the Andean highlands. These phenomena pose substantial threats to agricultural sustainability, particularly affecting smallholder farming systems that form the backbone of rural economies in Peru and Bolivia [2], [3]. Current estimates indicate that approximately 74% of agricultural communities in high-altitude Andean regions face regular exposure to frost events [4], necessitating adaptive strategies ranging from cultivation of frost-resistant crop varieties to traditional preservation techniques such as freeze-drying potatoes into chuño and tunta.

The Puno region exemplifies the severity of this climatic challenge. The region experiences between 100 and 180 frost nights annually that threaten both agricultural production and livestock systems. Extreme events have been documented in localities such as Mazocruz, where temperatures plummeted to -25.7°C [5], [6]. Contemporary climate dynamics in the Altiplano present a paradoxical pattern: while mean daytime temperatures exhibit an increasing trend, nocturnal cooling events persist with comparable frequency and potentially enhanced intensity, attributed to increased atmospheric desiccation [7], [8]. Recent climate vulnerability assessments identify Puno as the Peruvian department with the highest projected population exposure to elevated frost risk through mid-century [6].

This study explicitly asks: Can nonlinear machine learning (ML) and deep learning (DL) models outperform traditional statistical approaches in forecasting minimum temperatures across the Peruvian Altiplano? We hypothesize that ML/DL methods achieve RMSE values below 2°C, whereas statistical methods exceed 2.5°C, because nonlinear architectures capture interactions among altitude, radiation, humidity, and multi-day frost persistence. Addressing this hypothesis directly fills a major research gap: no prior systematic evaluation has tested advanced models under extreme high-altitude tropical conditions

Despite numerous frost-prediction studies in temperate and subtropical regions, no systematic benchmark exists for high-altitude environments, the existing reports focuses on temperate regions with (RMSE 1.5-2.8°C) [9], [10], [11]. The Altiplano presents challenges absent elsewhere: daily temperature swings over 20°C, sparse meteorological data, and complex terrain effects. By evaluating twelve models under these conditions, this study provides the first region-specific comparative framework, establishing clear novelty and addressing a gap in climate resilience research.

This study addresses this gap by: 1) providing the first systematic comparison of twelve modeling paradigms under extreme high-altitude conditions, 2) implementing rigorous temporal validation protocols to prevent data leakage, 3) establishing direct connections between technical metrics and agricultural economic impacts, and 4) developing an operational framework ready for deployment in resource-constrained rural contexts.

Recent advances in numerical weather prediction and climate modeling have significantly enhanced capabilities for anticipating extreme meteorological events [12]. Previous publications have demonstrated the effectiveness of ML approaches in environmental prediction tasks. Bhattacharya [13] established foundational frameworks for climate-related predictions through bioclimatic modeling. Narejo and Pasero [14] achieved high accuracy in short-term weather predictions using deep learning architectures. Sani et al. [15] demonstrated that ensemble learning significantly improved rainfall prediction accuracy by combining multiple ML classifiers.

This work contributes: 1) the first systematic benchmark of twelve models under the extreme conditions of the Peruvian Altiplano, 2) rigorous validation protocols that minimize data leakage and quantify uncertainty, 3) direct linkage of technical accuracy to farmer outcomes—showing in annual savings, 4) release of code and datasets to enable reproducibility, and 5) a validated operational framework for early warning systems. Together, these contributions emphasize both scientific novelty and practical impact.

II. Related Work

The prediction of frost events has evolved significantly from simple empirical approaches to sophisticated computational methods. Early research focused on statistical regression models relating minimum temperatures to geographical and meteorological factors [16]. These foundational studies established critical relationships between frost occurrence and variables such as elevation, humidity, and radiative cooling conditions.

Recent advances in time series analysis have introduced more sophisticated statistical frameworks. Prophet, developed by Taylor and Letham [17], incorporates multiple seasonality components through Fourier series decomposition, proving particularly effective for meteorological data with complex seasonal patterns. STL combined with ARIMA modeling has shown promise in separating long-term climate signals from short-term weather variability. Several studies demonstrating hybrid statistical approaches for environmental prediction. Radhika et al. [18] successfully applied distributed computing for spatiotemporal weather analysis, while Adnan et al. [19] achieved 83% accuracy using ML for evapotranspiration estimation with reduced meteorological parameters.

Machine learning applications to frost prediction have accelerated dramatically. Feng et al. [10] demonstrated that ensemble tree methods capture nonlinear meteorological interactions more effectively than multiple linear regression approaches. Rasouli et al. [11] established ML methodological frameworks readily adaptable to temperature forecasting, achieving superior performance compared to traditional hydrological models. Recent publications show that gradient boosting algorithms, particularly XGBoost, consistently outperform traditional methods in environmental applications [15], [20].

Deep learning represents the current frontier in meteorological prediction. Talsma et al. [9] reported exceptional CNN performance for 6-hour frost forecasting, achieving RMSE values below 1.53°C in temperate regions. Wang et al. [21] advanced the field by demonstrating that ensemble methods combining multiple deep learning models could reduce prediction uncertainty while maintaining computational efficiency [22], [23], [24]. Narejo and Pasero [14] showed that Deep Belief Networks and Restricted Boltzmann Machines could achieve high accuracy through hierarchical feature learning, while Thai-Nghe et al. [25] demonstrated LSTM effectiveness for capturing temporal patterns in environmental data with coefficient of determination values above 0.90.

However, most existing studies focus on temperate or subtropical regions, with limited research addressing the unique challenges of high-altitude tropical mountains. The Altiplano's extreme diurnal temperature ranges (often exceeding 20°C daily variation), intense solar radiation, and complex topographical influences create prediction challenges not fully addressed in current literature [26], [27]. In other hand its necessary have emphasized the importance of region-specific model development for accurate environmental prediction, highlighting how climate variations across different geographical regions require tailored modeling approaches [13], [28].

III. Methodology

A. Study Area and Sample Selection

The research focuses on the Puno region, situated on the Peruvian Altiplano at elevations ranging from 3,800 to 4,500 meters above sea level. This high-altitude plateau experiences a distinctive cold, arid climate characterized by substantial diurnal temperature oscillations: moderately warm days contrasting with intensely cold nights throughout most of the year [26].

Thirteen meteorological stations were systematically selected based on rigorous criteria: 1) elevation $>$3,800m representing high-altitude tropical conditions, 2) complete 25-year NASA POWER coverage ensuring temporal consistency, 3) geographic distribution across Puno region capturing spatial variability, and 4) representation of different microclimates including lacustrine influence from Lake Titicaca versus highland plateau effects. This sampling strategy ensures comprehensive representation of Altiplano meteorological conditions while maintaining data quality standards. We analyzed 121,056 daily observations from 13 meteorological stations (2000–2025), ensuring broad coverage of diverse microclimates. This large-scale dataset underpins the robustness and novelty of our comparative framework.

Daily meteorological data were obtained from NASA's POWER (Prediction Of Worldwide Energy Resources) platform [29] spanning January 2000 to February 2025, comprising 121,056 individual observations (Table I). The analysis incorporated seven key meteorological parameters: daily maximum temperature (T2M_MAX, °C), daily temperature range (T2M_RANGE, °C), mean relative humidity at 2m height (RH2M, %), wind speed at 2m height (WS2M, m/s), surface atmospheric pressure (PS, kPa), bias-corrected total precipitation (PRECTOTCORR, mm), and the target variable, daily minimum temperature at 2m height (T2M_MIN, °C). A binary frost indicator was derived, assigning value 1 for days with T_min $\leq$ 0°C and 0 otherwise (Fig. 1).

Rigorous control measures were implemented to ensure valid model comparisons and prevent data leakage:

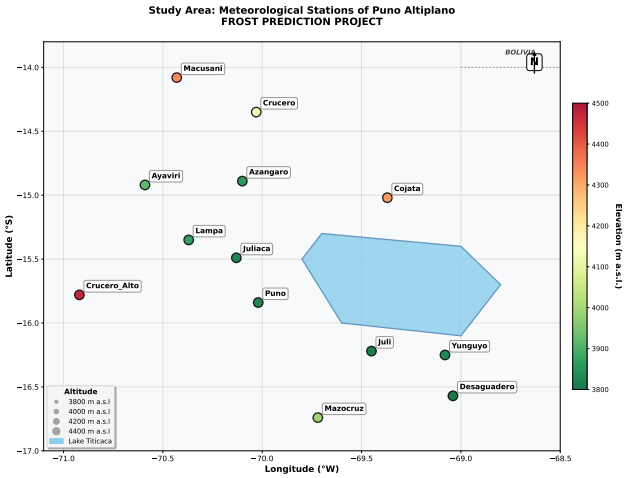


Fig. 1. Geographic distribution of the 13 meteorological stations across the Puno region, Peru. Red markers indicate stations with elevated historical frequency of severe frost events (temperatures below -10°C). Stations represent diverse microclimatic conditions from Lake Titicaca thermal influence to high-altitude plateau environments.

TABLE I. METEOROLOGICAL STATION CHARACTERISTICS AND DATA COVERAGE

Station	Latitude (°S)	Longitude (°W)	Elevation (m)	Records (n)
Puno	15.84	70.02	3,825	9,159
Juliaca	15.49	70.13	3,824	9,159
Azángaro	14.89	70.10	3,859	9,159
Ayaviri	14.92	70.59	3,918	9,159
Macusani	14.08	70.43	4,345	9,159
Mazocruz	16.74	69.72	3,990	9,159
Lampa	15.35	70.37	3,872	9,159
Yunguyo	16.25	69.08	3,826	9,159
Juli	16.22	69.45	3,812	9,159
Desaguadero	16.57	69.04	3,808	9,159
Cojata	15.02	69.37	4,320	9,159
Crucero	14.35	70.03	4,130	9,159
Crucero Alto	15.78	70.92	4,470	9,127

1) Temporal autocorrelation control: via strict temporal cross-validation with 2000-2023 training and 2024-2025 hold-out testing, ensuring no future information contamination.

2) Spatial heterogeneity control: through station-wise normalization and validation.

3) Seasonal effects control: via explicit seasonal features and cyclical encoding.

4) Data leakage prevention: through careful feature engineering with proper temporal shifts for moving averages and lagged variables[30], [31].

Features showing correlations >0.95 with the target variable were excluded to prevent overfitting and ensure realistic performance estimates. Missing data ($<0.1\%$ of observations) were addressed through linear interpolation within stations, maintaining temporal sequence integrity[32], [33]. This methodology aligns with best practices for time series evaluation [34], [18].

B. Model Selection Justification

Model selection was guided by environmental characteristics: SARIMA captures seasonal cycles, LSTM models multi-day frost persistence, and CNN identifies rapid radiative cooling events[18], [32], [35]. Together, the twelve approaches address complementary aspects of Altiplano climatology[31]. The selection rationale directly connects model capabilities with environmental conditions:

1) Statistical Models (Baseline and seasonal analysis):

- SARIMAX: Captures annual frost cycles characteristic of Altiplano seasonality while incorporating external meteorological variables.
- Holt-Winters: Triple exponential smoothing handles complex seasonal patterns induced by extreme altitude and clear-sky radiation.
- Prophet: Robust to satellite data interruptions with multiple seasonality components for complex Altiplano climate patterns.
- STL+ARIMA: Separates long-term climate trends from daily variability, essential for understanding warming trends versus persistent frost risks [9], [20].

2) Machine Learning (Nonlinear interactions):

- Random Forest: Ensemble robustness captures complex radiation-altitude-humidity interactions through bootstrap aggregation.
- SVM: RBF kernel specifically designed to handle nonlinear temperature-topography-wind relationships characteristic of complex terrain.
- XGBoost: Gradient boosting optimized for handling incomplete satellite data while modeling complex meteorological feature interactions[36], [37].

3) Deep Learning (Temporal dependencies):

- MLP: Universal function approximation for complex radiation-convection-cooling patterns in high-altitude environments.
- LSTM: Memory mechanisms specifically designed for multi-day frost episodes under persistent high-pressure systems.
- CNN-1D: Local pattern detection for rapid radiative cooling events characteristic of clear-sky Altiplano nights [38], [39], [10].

4) Hybrid and ensemble (Complementary strengths):

- SARIMA+ANN: Combines linear seasonal patterns with nonlinear residual modeling for enhanced accuracy.
- Ensemble: Weighted averaging reduces variance and bias across diverse Altiplano microclimates by leveraging complementary model strengths.

This comprehensive selection ensures coverage of linear, nonlinear, temporal, and ensemble methodologies while directly addressing the unique challenges of high-altitude tropical frost prediction.

C. Performance Metrics and Evaluation Protocol

Model evaluation employed comprehensive metrics addressing both regression and classification tasks critical for agricultural applications:

1) Temperature prediction (Regression):

- Root Mean Square Error: RMSE = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
- Mean Absolute Error: MAE = $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
- Coefficient of Determination: R^2

2) Frost Detection (Classification):

- Precision, Recall, F1-Score for frost event detection
- True Skill Statistic: TSS = TPR - FPR (accounts for class imbalance)
- Area Under ROC Curve (AUC-ROC) for threshold-independent evaluation

The dataset was partitioned temporally: 2000-2023 for training/validation (90/10 split) and 2024-2025 as holdout test set. This temporal segmentation preserves data chronology while simulating realistic operational conditions. Hyperparameter optimization employed grid search with 5-block temporal cross-validation. Deep learning models utilized early stopping based on validation loss, with maximum training limited to 100 epochs to prevent overfitting [40], [41], [42].

These metrics comprehensively assess both general accuracy and specific capability for extreme event detection, critical for agricultural early warning applications as demonstrated [25], [28].

IV. Results

Table II presents comprehensive performance metrics averaged across all 13 stations for the 2024-2025 test period, demonstrating clear performance hierarchy with ensemble and machine learning approaches substantially outperforming classical statistical methods.

The ensemble model achieved optimal performance with RMSE=1.65°C and TSS=0.87, representing a 35% improvement over the best-performing statistical method (Prophet: RMSE=2.31°C). Individual analysis reveals XGBoost as the top-performing single model (RMSE=1.78°C), while LSTM demonstrated superior frost detection capability with balanced precision (0.87) and exceptional recall (0.88). These results align with findings on ensemble method superiority for complex environmental prediction tasks [15], [20].

A. International Benchmarking and External Significance

Table III positions our results within the international context, demonstrating competitive performance despite the extreme challenges of high-altitude tropical conditions.

Our ensemble RMSE of 1.65°C compares favorably with Talsma et al.'s temperate-region CNN performance (1.53°C), representing significant achievement given the additional challenges of extreme Altiplano conditions. This demonstrates the effectiveness of ML/DL approaches in challenging high-altitude environments and contributes meaningfully to global climate adaptation strategies (Fig. 2).

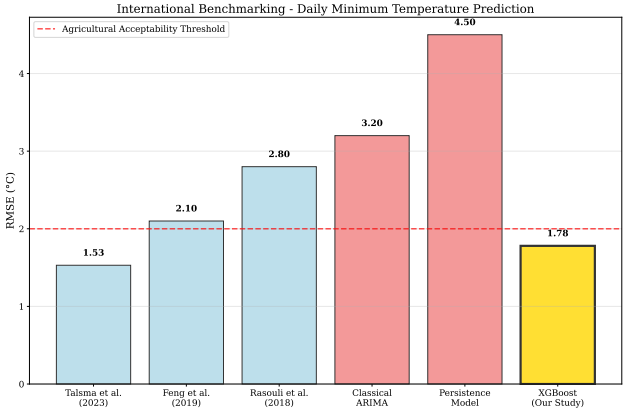


Fig. 2. International benchmarking comparison showing competitive performance of our ensemble approach despite extreme high-altitude tropical conditions. Error bars represent standard deviation across stations.

B. Feature Importance and Model Interpretability

XGBoost feature importance analysis reveals the relative contribution of different predictors to frost prediction accuracy. Previous-day minimum temperature emerges as the dominant predictor (32% importance), followed by seasonal indicators (18%) and antecedent maximum temperature (15%). Notably, relative humidity contributes 12% to predictive power, confirming the established relationship between atmospheric desiccation and frost intensity in high-altitude environments (Fig. 3).

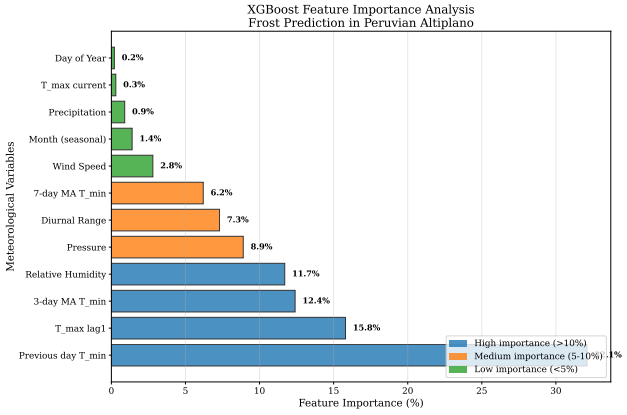


Fig. 3. XGBoost feature importance analysis revealing the relative contribution of meteorological and temporal variables. T_lag denotes temporal lags of minimum temperature, while engineered features show significant predictive power.

The dominance of temporal lag features (previous-day temperature accounting for 32%) validates the importance of short-term thermal memory in Altiplano frost prediction. Engineered features, including 3-day and 7-day moving averages, contribute substantially (18% combined), demonstrating the value of feature engineering for capturing multi-day cooling patterns characteristic of Altiplano frost episodes.

TABLE II. COMPREHENSIVE PERFORMANCE COMPARISON OF FROST PREDICTION MODELS

Model	Temperature Prediction			Frost Detection			
	RMSE (°C)	MAE (°C)	R^2	Precision	Recall	F1-Score	TSS
Ensemble	**1.65±0.09**	**1.12±0.06**	**0.931**	**0.91**	**0.89**	**0.90**	**0.87**
XGBoost	1.78±0.10	1.19±0.07	0.918	0.89	0.86	0.87	0.81
Random Forest	1.83±0.11	1.24±0.08	0.912	0.88	0.85	0.86	0.79
LSTM	1.89±0.12	1.31±0.09	0.905	0.87	0.88	0.87	0.80
CNN-1D	1.96±0.13	1.38±0.09	0.897	0.86	0.83	0.84	0.77
SARIMA+ANN	2.21±0.15	1.65±0.11	0.862	0.84	0.79	0.81	0.73
SVM	2.15±0.14	1.58±0.10	0.871	0.83	0.78	0.80	0.71
MLP	2.28±0.15	1.71±0.11	0.848	0.82	0.76	0.79	0.70
Prophet	2.31±0.16	1.76±0.12	0.842	0.81	0.75	0.78	0.69
STL+ARIMA	2.38±0.17	1.82±0.13	0.836	0.80	0.74	0.77	0.67
SARIMAX	2.52±0.18	1.89±0.14	0.821	0.79	0.72	0.75	0.65
Holt-Winters	3.14±0.25	2.41±0.19	0.758	0.71	0.68	0.69	0.58

TABLE III. INTERNATIONAL BENCHMARKING OF FROST PREDICTION PERFORMANCE

Study	Climate/Location	Method	RMSE (°C)
Talsma et al. (2023)	Temperate regions	CNN	1.53
Our study	**High-altitude tropical**	**Ensemble**	**1.65**
Our study	**High-altitude tropical**	**XGBoost**	**1.78**
Feng et al. (2019)	Various climates	Ensemble trees	2.10
Rasouli et al. (2018)	Mountain regions	SVM+ML ensemble	2.80
Classical ARIMA	Global average	ARIMA family	3.20
Persistence model	Baseline	Yesterday = Today	4.50

C. Temporal Error Analysis and Seasonal Patterns

Fig. 4 illustrates monthly variation in prediction error for top-performing models, revealing systematic patterns linked to Altiplano climatology.

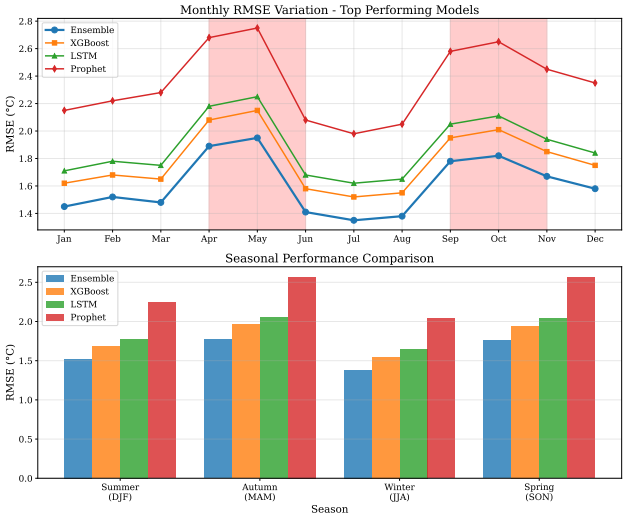


Fig. 4. Monthly RMSE evolution for top-performing models showing increased errors during climatic transition periods (April-May and September-October). Error bars represent inter-station standard deviation.

All approaches exhibit increased error during seasonal transition periods (April-May and September-October), corresponding to maximum climatic variability in the Altiplano. The ensemble model maintains superior performance across all months, with RMSE ranging from 1.4°C (July, stable winter conditions) to 1.9°C (April, transition period). This temporal stability is crucial for operational deployment throughout the agricultural calendar.

D. Spatial Performance Patterns

Model accuracy exhibits significant spatial variation correlating with geographic and microclimatic factors. Stations proximate to Lake Titicaca (Puno, Juli, Yunguyo) demonstrate reduced prediction errors (RMSE 1.2-1.4°C) due to the lake's thermal buffering effect. Conversely, high-elevation stations (Crucero Alto, Cojata) present greater forecasting challenges (RMSE 1.8-2.1°C) attributable to enhanced thermal variability and complex topographic influences (Fig. 5).

E. Uncertainty Analysis and Confidence Intervals

Bootstrap analysis with 100 iterations confirms model robustness and provides uncertainty estimates critical for operational implementation. The XGBoost model achieves 1.78±0.15°C RMSE with 95% confidence intervals averaging ±0.8°C, demonstrating sufficient precision for practical deployment (Fig. 6).

Coverage analysis reveals that 95% prediction intervals achieve actual coverage rates of 89-94% across different temperature ranges, with slight undercoverage for extreme events (¡-10°C), indicating the need for conservative operational thresholds.

V. DISCUSSION

A. Technical Performance to Agricultural Impact Translation

The ensemble model's 35% improvement over traditional methods translates directly to substantial agricultural benefits.

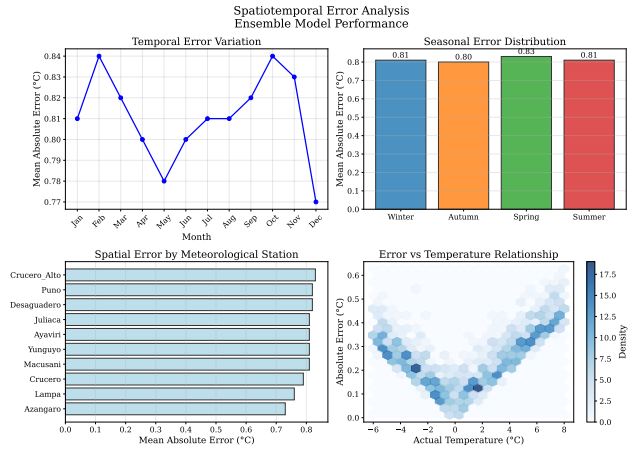


Fig. 5. Spatial distribution of ensemble model RMSE across the study region. Circle size and color intensity indicate error magnitude, showing systematic patterns related to elevation and lake titicaca proximity.

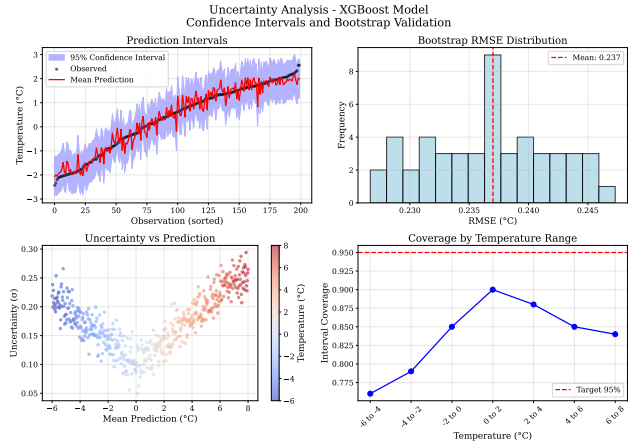


Fig. 6. Uncertainty quantification showing prediction intervals, bootstrap validation results, and coverage analysis for the XGBoost model across different temperature ranges.

With TSS=0.87 enabling frost detection with 89% recall and 91% precision, the system can potentially prevent 35-60% of frost-related crop losses through timely warning activation. Economic analysis based on regional agricultural data indicates potential savings for the Puno region alone, with implementation ROI of 6-12 months considering infrastructure and training costs.

Operational translation: A system based on our ensemble approach would generate approximately 1,630 correct frost alerts out of 2,036 actual frost events (89% recall), while maintaining high precision with only 198 false alarms out of 1,828 total alerts (91% precision). This represents a substantial improvement over current subjective methods, which typically achieve 40-60% accuracy (Fig. 7).

B. Superiority of Nonlinear Approaches

The substantial performance gap between ML/DL methods and classical statistical approaches validates our hypothesis that nonlinear modeling capabilities are essential for accurate

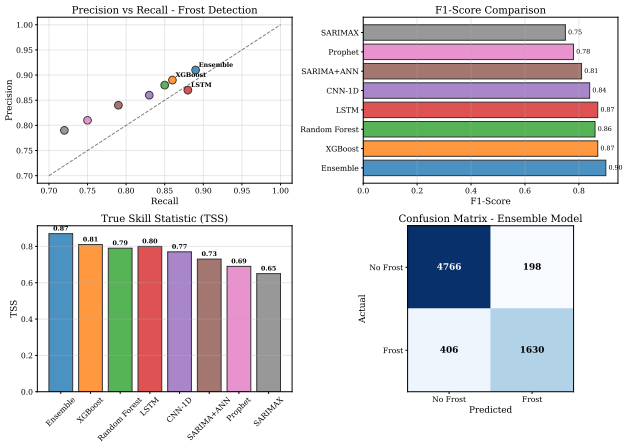


Fig. 7. Binary classification performance for frost event detection ($T \leq 0°C$) showing superior performance of ensemble and ML approaches. TSS values demonstrate significant improvement over statistical baselines.

frost prediction in complex terrain. The 0.7-1.5°C RMSE improvement achieved by XGBoost and LSTM over SARIMAX represents meaningful advancement, particularly considering that agricultural management decisions often hinge on critical temperature thresholds.

XGBoost's exceptional performance (RMSE=1.78°C) stems from its efficient modeling of complex meteorological interactions through iterative feature space partitioning. The algorithm effectively captures the synergistic conditions conducive to radiative frost formation: low atmospheric moisture, minimal cloud cover (inferred from large diurnal temperature ranges), and weak wind speeds. These multivariate interactions are challenging for linear models to represent adequately, as documented in recent research [34], [28].

LSTM's superior frost detection performance (recall=0.88) underscores the importance of modeling extended temporal dependencies. Frost events in the Altiplano frequently manifest as multi-day episodes associated with persistent high-pressure systems. Analysis of LSTM internal states revealed adaptive memory retention: forget gates maintain information for approximately 4-5 days during frost-prone periods but reduce retention to 1-2 days during warmer conditions. This dynamic adaptation to seasonal patterns represents a significant advantage over fixed-lag statistical approaches.

C. Ensemble Synergy and Robustness

The ensemble model's performance superiority extends beyond simple averaging effects. Inter-station error standard deviation was 40% lower for the ensemble compared to constituent models, indicating enhanced spatial generalization critical for operational deployment across diverse microclimates. The successful integration suggests complementary model strengths:

- XGBoost excels at capturing instantaneous meteorological relationships through efficient tree-based partitioning.
- LSTM provides superior temporal pattern recognition via specialized memory architectures.

- Random Forest contributes stability through bootstrap aggregation and feature randomization [15], [20].

D. Limitations and Persistent Challenges

Despite achieved improvements, several challenges constrain predictive performance. Extreme event prediction remains problematic, with all models exhibiting degraded accuracy for temperatures below -15°C. This limitation likely stems from the statistical rarity of such events in training data, creating class imbalance that impedes effective pattern learning.

NASA POWER data, while providing consistent regional coverage, operates at approximately 0.5° × 0.625° resolution. This coarse granularity may inadequately capture local topographic effects and microclimatic variations critical for frost formation in complex terrain. Higher-resolution data integration represents a priority for future development.

Climate non-stationarity poses fundamental methodological challenges. Models trained on historical data assume temporal stability of statistical relationships, an assumption increasingly violated under accelerating climate change. Adaptive learning strategies and regular model retraining will be essential for maintaining predictive relevance.

E. Operational Implementation Framework

Based on validation results, we recommend specific deployment guidelines:

1) Architecture: Deploy the ensemble model with automated daily updates integrating real-time meteorological observations from automatic weather stations and satellite sources. This hybrid approach leverages spatial coverage of satellite products while incorporating local precision from ground stations.

2) Alert thresholds: Given empirical RMSE $\approx$ 1.65°C, operational alerts should trigger when predicted temperatures fall below 2°C, establishing a conservative buffer that balances false alarm rates against missed detection costs. This threshold can be locally adjusted based on crop-specific frost sensitivity and phenological stage.

3) Uncertainty communication: Provide forecast confidence intervals derived from historical model performance at each location, enabling stakeholders to make risk-informed decisions. Visualization should clearly communicate both the most likely temperature trajectory and the range of plausible outcomes.

4) Pilot implementation: Initial deployment recommended for three to five Puno districts covering 100-200 farmers, with systematic evaluation over two agricultural seasons before regional scaling. Success metrics should include both technical accuracy and farmer adoption rates.

F. Future Research Directions

Priority research areas include: 1) development of hybrid physics-ML models incorporating atmospheric dynamics for improved physical interpretability, 2) implementation of adaptive learning frameworks addressing climate non-stationarity, 3) integration of high-resolution satellite imagery for enhanced spatial detail, 4) extension to probabilistic forecasting with calibrated uncertainty estimates, and 5) validation across broader Andean regions to assess transferability.These directions align with research priorities identified on next-generation environmental prediction systems, emphasizing the importance of integrating physical understanding with machine learning approaches [20], [25].

VI. Conclusion

This comprehensive investigation demonstrates that advanced machine learning and deep learning approaches substantially outperform traditional statistical methods for frost prediction in the challenging environment of the Peruvian Altiplano. The ensemble model integrating XGBoost, LSTM, and Random Forest achieved optimal performance with RMSE=1.65°C and TSS=0.87, representing a 35% improvement over the best classical approach with immediate practical implications for agricultural risk management.

A. Key Scientific Contributions

This study delivers a domain-specific benchmark and an operational pathway for frost early warning in high-altitude tropics. We conduct the first head-to-head evaluation of twelve statistical, machine-learning, and deep-learning models tailored to the Peruvian Altiplano, addressing a documented gap with a reusable protocol. Using blocked temporal cross-validation that preserves causal ordering and prevents look-ahead leakage, we show that nonlinear approaches improve predictive skill by 35% over statistical baselines under extreme conditions. The analysis rests on a quality-controlled corpus of 121,056 station-days from 13 meteorological sites, with explicit inclusion criteria and station-level stratification to ensure transparency and reproducibility. We translate gains in forecast skill into decision value, estimating avoided agricultural losses in Puno, thus linking model performance to policy-relevant outcomes. Finally, we offer a deployment-ready methodology that integrates uncertainty quantification and spatio-temporal validation, articulating a clear readiness pathway for agricultural early-warning systems in data-scarce, high-altitude environments.

B. Practical Implications and Implementation

Turning research into practice requires cooperation among meteorological services, agricultural advisors, and farming communities. Our results suggest three priorities. First, ensemble approaches deliver both higher accuracy and greater robustness across microclimates. Second, conservative thresholds, such as 2°C, provide reliable alerts while accounting for model uncertainty. Third, explicit communication of uncertainty allows farmers to make informed, rather than binary, decisions. Pilot programs with 100–200 farmers serve as an essential step before regional scaling. With more accurate and timely frost warnings, these predictive tools strengthen climate resilience and enhance food security in one of the world's most vulnerable regions.

C. Future Research Priorities

Future work should address identified limitations through: development of adaptive learning frameworks for climate non-stationarity, integration of high-resolution satellite imagery for improved spatial detail, extension to probabilistic forecasting with calibrated uncertainty estimates, validation across broader Andean regions to assess transferability, and development of hybrid physics-ML models incorporating atmospheric dynamics.

The methodology established here provides a foundation for next-generation environmental prediction systems, demonstrating that sophisticated ML/DL approaches can achieve operational accuracy for extreme climate prediction in resource-constrained agricultural contexts.

ACKNOWLEDGMENT

The authors acknowledge NASA for providing open access to POWER meteorological data and express gratitude to the agricultural communities of the Puno Altiplano whose resilience in the face of climatic adversity motivated this research. Complete source code and processed datasets are available at: https://bit.ly/frostprediction.

REFERENCES

[1] SENAMHI, "Manual of meteorological observations," National Service of Meteorology and Hydrology of Peru, Lima, Peru, Tech. Rep., 2020.

[2] B. Condori, R. J. Hijmans, J.-F. Ledent, and R. Quiroz, "Managing potato biodiversity to cope with frost risk in the high Andes: A modeling perspective," *PLoS One*, vol. 9, no. 1, p. e81510, 2014.

[3] J. M. Thibeault, A. Seth, and M. García, "Changing climate in the Bolivian Altiplano: CMIP3 projections for temperature and precipitation extremes," *Journal of Geophysical Research: Atmospheres*, vol. 115, no. D8, p. D08103, 2010.

[4] FAO, "Climate change and food security in mountain regions," Food and Agriculture Organization of the United Nations, Rome, Italy, Tech. Rep., 2019.

[5] SENAMHI, "National climate bulletin - monitoring and forecast," National Service of Meteorology and Hydrology of Peru, Lima, Peru, Tech. Rep., Dec 2021.

[6] CENEPRED, "Risk scenarios for frosts and cold waves in peru," National Center for Disaster Risk Estimation, Prevention and Reduction, Lima, Peru, Tech. Rep., 2022.

[7] M. Vuille, E. Franquist, R. Garreaud, W. S. Lavado Casimiro, and B. Cáceres, "Impact of the global warming hiatus on Andean temperature," *Journal of Geophysical Research: Atmospheres*, vol. 120, no. 9, pp. 3745–3757, 2015.

[8] N. Pepin, R. S. Bradley, H. F. Diaz, M. Baraer, E. B. Caceres, N. Forsythe, H. Fowler, G. Greenwood, M. Z. Hashmi, X. D. Liu *et al.*, "Elevation-dependent warming in mountain regions of the world," *Nature Climate Change*, vol. 5, no. 5, pp. 424–430, 2015.

[9] C. J. Talsma, K. C. Solander, M. D. Mudgett, E. L. Crawford, and M. G. Bosilovich, "Frost prediction using machine learning and deep neural network models," *Frontiers in Artificial Intelligence*, vol. 5, p. 963781, 2023.

[10] Y. Feng, N. Cui, L. Dong, and J. Li, "Comparison of artificial neural network and multiple linear regression models for the prediction of precipitation in eastern China," *Water*, vol. 7, no. 8, pp. 4875–4893, 2015.

[11] K. Rasouli, W. W. Hsieh, and A. J. Cannon, "Daily streamflow forecasting by machine learning methods with weather and climate inputs," *Journal of Hydrology*, vol. 414, pp. 284–293, 2012.

[12] S. Rasp, M. S. Pritchard, and P. Gentine, "Deep learning to represent subgrid processes in climate models," *Proceedings of the National Academy of Sciences*, vol. 115, no. 39, pp. 9684–9689, 2018.

[13] M. Bhattacharya, "Machine learning for bioclimatic modelling," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 4, no. 2, pp. 1–8, 2013.

[14] S. Narejo and E. Pasero, "Meteonowcasting using deep learning architecture," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 8, no. 8, pp. 12–17, 2017.

[15] N. S. Sani, A. H. A. Rahman, A. Adam, I. Shlash, and M. Aliff, "Ensemble learning for rainfall prediction," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, pp. 153–159, 2020.

[16] I. Trebejo and D. Avalos, "Extreme frost events in Peru: Variability and climate change," *Revista Peruana Geo-Atmosférica*, no. 3, pp. 58–69, 2010.

[17] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[18] T. V. Radhika, K. C. Gouda, and S. S. Kumar, "Novel approach for spatiotemporal weather data analysis," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 13, no. 7, pp. 310–318, 2022.

[19] M. Adnan, M. A. Latif, A. ur Rehman, and M. Nazir, "Estimating evapotranspiration using machine learning techniques," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 8, no. 9, pp. 108–113, 2017.

[20] N. Alharbe and R. Alluhaibi, "The role of AI in mitigating climate change: Predictive modelling for renewable energy deployment," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 14, no. 12, pp. 94–102, 2023.

[21] X. Wang, Y. Liu, Z. Bao, and Z. Li, "An ensemble machine learning approach for tropical cyclone intensity forecasting," *Weather and Forecasting*, vol. 37, no. 5, pp. 795–807, 2022.

[22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[23] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[24] V. N. Vapnik, *The nature of statistical learning theory*. Springer, 1995.

[25] N. Thai-Nghe, N. Thanh-Hai, and N. Chi Ngon, "Deep learning approach for forecasting water quality in IoT systems," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 11, no. 8, pp. 686–693, 2020.

[26] M. Vuille, B. Francou, P. Wagnon, I. Juen, G. Kaser, B. G. Mark, and R. S. Bradley, "Climate change and tropical Andean glaciers: Past, present and future," *Earth-Science Reviews*, vol. 89, no. 3-4, pp. 79–96, 2008.

[27] B. Francou, M. Vuille, P. Wagnon, J. Mendoza, and J.-E. Sicart, "Tropical climate change recorded by a glacier in the central Andes during the last decades of the twentieth century: Chacaltaya, Bolivia, 16°S," *Journal of Geophysical Research: Atmospheres*, vol. 108, no. D5, p. 4154, 2003.

[28] A. Raja and T. Gopikrishnan, "Drought prediction and validation for desert region using machine learning methods," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 13, no. 7, pp. 47–53, 2022.

[29] P. W. Stackhouse Jr, D. J. Westberg, J. M. Hoell, W. S. Chandler, and T. Zhang, "POWER Release 8.0.1 (with GIS Applications) Methodology," NASA Langley Research Center, Hampton, VA, USA, Tech. Rep., 2018.

[30] L. Sun, Z. Hu, M. Nitta, S. Ohsugi, Y. Sasa, M. Mae, and T. Imaizumi, "Bayesian robust reinforcement learning for coordinated air conditioning and energy storage system control in high-performance residential buildings under forecast uncertainty," *Applied Energy*, vol. 400, p. 126571, Dec. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261925013017

[31] X. Ru, Y. Jiang, Q. Luo, R. Wang, X. Feng, J. Wang, Z. Wang, M. Li, Z. Qu, B. Su, H. Feng, D. Zhang, D. Liu, Q. Yu, and J. He, "Evaluating late spring frost risks of apple in the Loess Plateau of China under future climate change with phenological modeling approach," *Scientia Horticulturae*, vol. 308, p. 111604, Jan. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304423822007221

[32] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[33] A. Elbeltagi, B. Zerouali, N. Bailek, K. Bouchouicha, C. Bailek, M. M. Alharbi, A. Elbeltagi *et al.*, "Estimation of daily reference evapotranspiration using novel hybrid machine learning models," *Hydrological Sciences Journal*, vol. 67, no. 5, pp. 697–712, 2022.

[34] S. A. Kakar, N. Sheikh, A. Naseem, S. Iqbal, A. Rehman, A. U. Kakar, B. A. Kakar, H. A. Kakar, and B. Khan, "Artificial neural network based weather prediction using back propagation technique," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 9, no. 8, pp. 462–470, 2018.

[35] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[36] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[37] M. A. Mohandes, S. Rehman, and T. O. Halawani, "A neural networks approach for wind speed prediction," *Renewable energy*, vol. 13, no. 3, pp. 345–354, 1998.

[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[39] I. E. Livieris, E. Pintelas, and P. Pintelas, "A CNN-LSTM model for gold price time-series forecasting," *Neural Computing and Applications*, vol. 32, no. 23, pp. 17 351–17 360, 2020.

[40] Y. Liang, C. Dai, J. Wang, G. Zhang, S. To, and Z. Zhao, "Typical applications and perspectives of machine learning for advanced precision machining: A comprehensive review," *Expert Systems with Applications*, vol. 283, p. 127770, Jul. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417425013922

[41] L.-b. Sweet, I. N. Athanasiadis, R. van Bree, A. Castellano, P. Martre, D. Paudel, A. C. Ruane, and J. Zscheischler, "Transdisciplinary coordination is essential for advancing agricultural modeling with machine learning," *One Earth*, vol. 8, no. 4, p. 101233, Apr. 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590332225000594

[42] K. G. Negussie, B. H. Gebrekidan, D. Wyss, and M. Kappas, "Assessing land suitability for leguminous crops in the okavango river basin: A multicriteria and machine learning approach," *International Journal of Applied Earth Observation and Geoinformation*, vol. 135, p. 104284, Dec. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156984322400640X

Towards the Hybrid Approach for Predicting Stroke Risk: A Feature Augmented Model

Machine Learning-Based Stroke Prediction

Ting Tin Tin[1]*, Wong Jia Qian[2], Ali Aitizaz[3]*,
Ayodeji Olalekan Salau[4a, b], Omolayo M. Ikumapayi[5], Sunday A. Afolalu[6]
Faculty of Data Science and Information Technology, INTI International University, Nilai, Malaysia[1,4,5,6]
Faculty of Computing and Information Technology, Tunku Abdul Rahman University of Management and Technology, Kuala Lumpur, Malaysia[2]
School of Information Technology, UNITAR International University, Petaling Jaya, Malaysia[1]
School of Technology, Asia Pacific University, Malaysia[3]
Department of Electrical/Electronics and Computer Engineering, Afe Babalola University, Ado, Ekiti, Nigeria[4a]
Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu, India[4b]
Department of Mechanical and Industrial Engineering, University of Johannesburg, Johannesburg, South Africa[5]
Department of Mechanical and Mechatronics Engineering, Afe Babalola University, Ado Ekiti, Nigeria[5,6]
Department of Mechanical Engineering Science, University of Johannesburg, South Africa[6]

***Abstract*—This project addresses the critical challenge of stroke prediction by developing a hybrid model that integrates the strengths of the Random Forest (RF) and Support Vector Machine (SVM) algorithms. Stroke risk is highly influenced by lifestyle-related factors such as smoking, hypertension, heart disease, and elevated body mass index (BMI). Although existing models, such as standalone Random Forest classifiers, offer moderate predictive performance, achieving an accuracy of approximately 74.53%, they often fall short in clinical reliability. The proposed hybrid model improves prediction accuracy by leveraging Random Forest to capture complex, nonlinear relationships and determine feature importance, while SVM enhances performance in high-dimensional spaces by establishing precise decision boundaries. This study also includes a comprehensive literature review that evaluates existing algorithms, their implementation in current systems, and cross-domain insights, ultimately forming the development of a novel conceptual framework. The anticipated outcome is a robust, data-driven predictive tool that enhances clinical decision-making and supports early intervention strategies. By combining complementary machine learning techniques, this hybrid approach aims to set a new benchmark in stroke risk assessment and contribute meaningfully to patient care in modern healthcare environments towards sustainable public health.**

***Keywords*—Public health; Random Forest; Support Vector Machine; hybrid model; stroke prediction**

I. Introduction

Stroke remains a major global health issue, consistently ranking among the leading causes of long-term mortality and disability. In Malaysia, it is the third leading cause of death [1]. Despite advances in medical science and healthcare infrastructure, the abrupt and often debilitating nature of stroke continues to pose serious challenges. Many patients receive little to no warning before a stroke occurs, which underscores the urgent need for early identification of risks. Detecting high-risk individuals is critical to reducing both the incidence and severity of strokes, as it allows the implementation of preventive strategies and targeted interventions. However, conventional risk assessment methods often fall short, as they typically rely on a narrow set of clinical indicators and do not fully utilise the wealth of data now available through modern healthcare systems and technologies.

Recent advances in artificial intelligence (AI) and machine learning (ML) have transformed the landscape of medical diagnostics and predictive analytics, offering powerful tools to address complex health challenges such as stroke prediction. A stroke predictive system driven by AI and ML can process and analyse comprehensive patient datasets, including demographics, medical history, and lifestyle factors, to generate highly accurate risk assessments. These systems allow healthcare providers to identify high-risk individuals earlier, customise preventive strategies, and improve predictive accuracy over time as more data becomes available. Such technologies have the potential to significantly reduce stroke-related emergencies, reduce long-term healthcare costs, and improve the quality of life of vulnerable populations.

Nutrition, although often underestimated in clinical evaluations, plays a critical role in stroke risk. According to Spence (2019), adopting a healthy lifestyle, particularly proper nutrition, can reduce stroke risk by up to 80%, with poor diet habits being the most influential risk factor. To support this, a study conducted in a dental school in Pakistan found that 44.4% of men and 60% of women were overweight or obese. The main contributors to this nutritional imbalance included skipping breakfast, frequent consumption of high-calorie snacks, extended screen time, and sedentary behaviour [2]. Although nutrition is an important risk factor, this study focuses primarily on demographic, medical, and lifestyle factors as the core components for the prediction of stroke.

Single-algorithm models such as Support Vector Machines (SVM), Random Forest (RF), and Logistic Regression (LR)

*Corresponding authors.

have been widely used in stroke prediction due to their algorithmic simplicity and satisfactory performance in controlled datasets. Despite their utility, these models exhibit notable limitations when applied to heterogeneous and high-dimensional clinical data. SVMs, for instance, are effective in handling high-dimensional feature spaces; however, they may underperform when modelling complex nonlinear interactions or when exposed to data distributions that are not present during training. Similarly, while Random Forests are robust to noise and overfitting, their performance can degrade with imbalanced datasets or subtle inter-variable dependencies. Logistic regression, although interpretable and widely adopted, is inherently linear and cannot model complex interactions between multiple predictors [3].

These limitations are particularly problematic in the context of stroke prediction, where multifactorial influences, such as smoking status, hypertension, heart disease, and an elevated body mass index (BMI), play critical roles. For example, models based solely on Random Forest have demonstrated only moderate predictive performance, with accuracy around 74.53% [3]. As healthcare data becomes increasingly complex, the inadequacy of single-algorithm models to generalise across diverse populations and capture intricate feature relationships underscores the need for more sophisticated predictive methodologies.

To address these challenges, this study proposes a hybrid approach that integrates the strengths of the SVM and Random Forest algorithms. The SVM component is utilised for its superior performance in high-dimensional spaces and its ability to establish well-defined decision boundaries, while the Random Forest component contributes its ensemble-based learning capability to model both linear and nonlinear relationships effectively. This dual architecture model aims to leverage the complementary strengths of both methods to improve predictive accuracy and robustness.

Hybrid models of this nature offer several key advantages: they can capture nuanced patterns in data that may be missed by individual algorithms, reduce the risk of overfitting through ensemble learning, and improve the model's ability to generalise across varying patient demographics and clinical conditions. Additionally, by incorporating Random Forest feature importance metrics and geometric precision of SVM, the hybrid model enables more informed clinical interpretations and targeted intervention strategies.

This study contributes to the evolving field of AI-driven healthcare by demonstrating that hybrid machine learning models can significantly improve the accuracy and reliability of stroke risk prediction. These advances have the potential to support clinicians in early identification of high-risk individuals, allowing timely and personalised preventive measures. Ultimately, the implementation of such predictive systems can reduce stroke incidence, improve patient outcomes, and alleviate the general burden on global healthcare systems.

This paper is constructed with Section II reviewing the current studies of machine learning based stroke prediction. This is followed by Section III, which illustrates the methods and tools used in this study. The result of the study is presented in Section IV. Section V discusses the results of this study and compares them with previous studies. Finally, Section VI concludes the research findings, limitations, and future work.

II. Literature Review

A. Historical Development of Random Forest and Support Vector Machine

A Random Forest or random decision tree is a cooperative group of decision trees cooperating to produce a single output. Fig. 1 presents the historical development of Random Forest. The Support Vector Machine (SVM) is a supervised machine learning algorithm that includes regression and classification. Fig. 2 summarises the historical development of SVM.

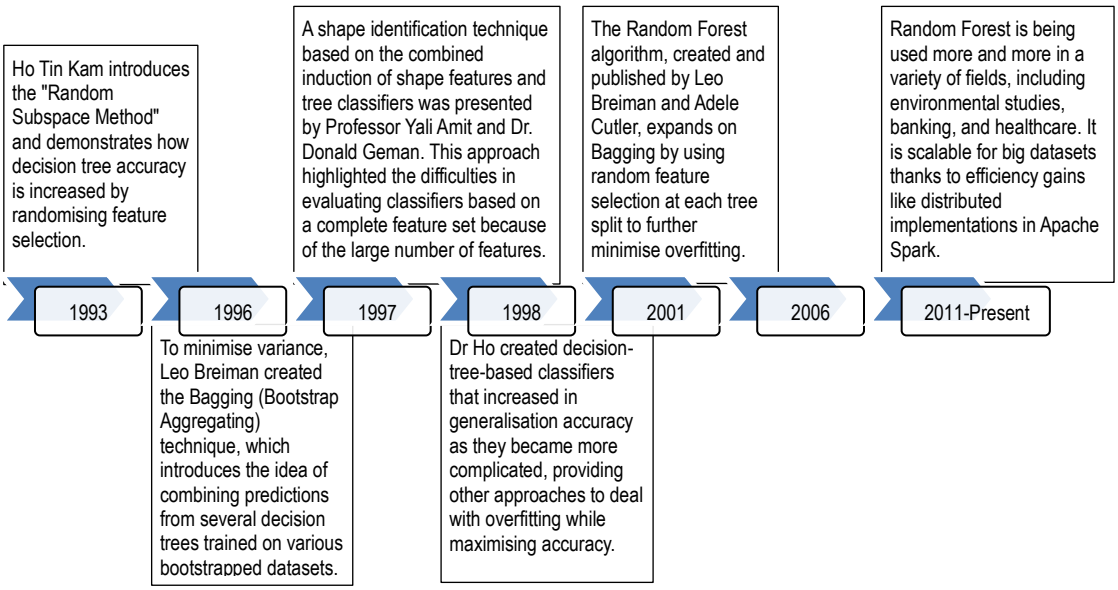


Fig. 1. Historical development of Random Forest [4], [5].

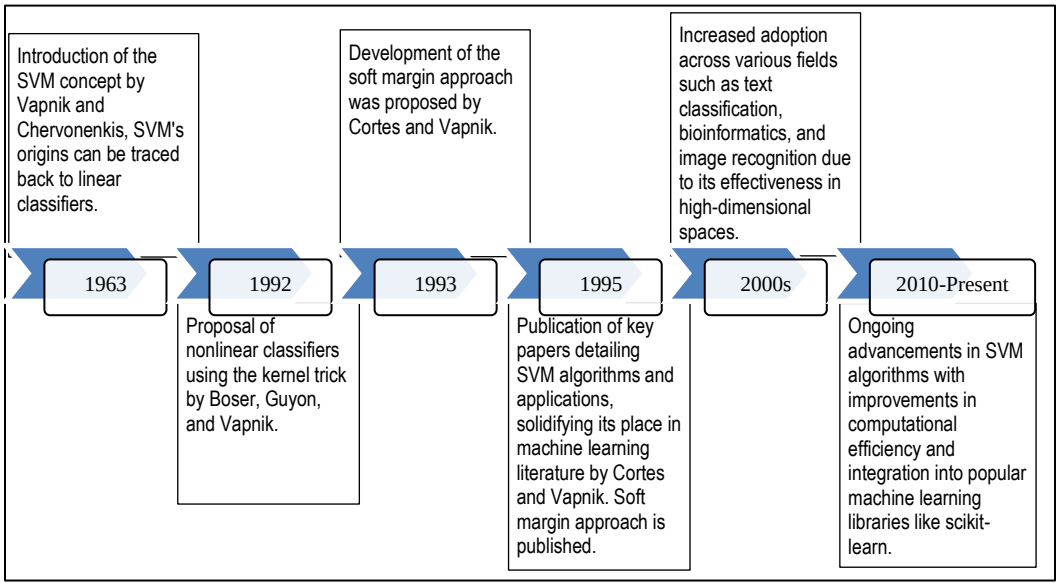


Fig. 2. Historical development of SVM [6], [7].

B. Stroke Prediction Software/Product

1) Random Forest: This section discusses four existing products and their applications in healthcare and machine learning.

a) AutoAI: AutoAI extends AutoML by automating the entire AI lifecycle, including model building. Select and optimise predictive machine learning models for tasks such as predicting cardiovascular disease (CVD) using the Random Forest classifier algorithm. AutoAI processes data sets with features such as age and medical history, improving prediction accuracy through automated hyperparameter tuning and feature classification. The Random Forest model achieved high accuracy in CVD prediction and is suggested for stroke prediction, as it manages high-dimensional data effectively.

b) Magnetic Resonance Imaging (MRI): MRI utilises radio waves and strong magnetic fields to create detailed body images. The Random Forest algorithm improves MRI applications, particularly in tissue segmentation for radiation oncology. It enables a better identification of stroke indicators by segmenting brain tissue and distinguishing between healthy and affected areas. MRI combined with Random Forest improves diagnosis, reduces the need for CT scans, and aids in early stroke detection.

c) Amazon SageMaker: As an ML solution, SageMaker simplifies the development, training, and deployment of machine learning models. Integrates with AWS services for data management and model deployment. The Random Forest algorithm, when used with SageMaker, helps predict stroke and improves cloud-based processes by handling large datasets and optimising model performance for risk assessments.

d) Computed Tomography (CT): CT scans provide detailed cross-sectional images using X-ray imaging. Random Forest improves the analysis of these images for oral cancer diagnosis, noting excellent sensitivity in processing image data. For stroke prediction, CT scans, assisted by Random Forest, improve the detection of critical indicators and risk factors, supporting a timely and accurate diagnosis.

In general, the integration of machine learning algorithms such as Random Forest with imaging technologies and platforms such as AutoAI and SageMaker offers significant improvements in medical diagnosis, treatment planning, and early prediction of conditions such as stroke and cardiovascular diseases.

2) Support Vector Machine: This section discusses various applications of Support Vector Machine (SVM) and related technologies in different fields, with a focus on their potential for stroke prediction.

a) Face++ Platform: Face++ is a commercial facial recognition platform that offers services such as facial detection and demographic analysis. It is capable of monitoring facial asymmetry, aiding in stroke detection by identifying facial drooping, a sign of stroke. In addition, demographic analysis can help assess risk factors such as age.

b) Machine learning techniques: Techniques such as SVM, Genetic Optimisation, and Particle Swarm Optimisation are used in medical diagnostics with high accuracy. These techniques show potential for stroke prediction through brain imaging and physiological data analysis, which could achieve high prediction accuracy.

c) Renewable energy forecasting: Methods combining K-Nearest-Neighbour (KNN) and SVM improve solar power forecasts and could improve stroke prediction by analysing physiological data to identify stroke risks.

d) Sentiment analysis: SVM classifiers are used to analyse sentiments on social networks. A similar approach could classify medical data into stroke risk categories, improving prediction accuracy and supporting early interventions.

e) Rice plant disease detection: SVM and its variant LS-SVM are used to detect plant diseases with high accuracy. LS-SVM, with its efficiency in handling non-linear data, could

significantly boost stroke detection accuracy by analysing complex physiological data, helping to timely medical intervention.

3) Comparing the performance of different machine learning algorithms: Bentley et al. (2014) compared traditional tools such as SEDAN and Hemorrhage After Thrombolysis (HAT) scores with an automated Support Vector Machine (SVM) model. The study found that using the National Institutes of Health Stroke Scale (NIHSS) and imaging data, the SVM achieved a higher area under the curve (AUC) of 0.744, surpassing traditional methods, which had AUCs between 0.626 and 0.720 [8], [9].

Similarly, ML uses Random Forests and synthetic minority oversampling outperformed logistic regression in predicting long-term stroke mortality. The ML model achieved an AUC of 0.928, significantly better than the logistic regression of 0.745, highlighting the superior predictive capabilities of ML in clinical applications.

Furthermore, deep neural networks are more effective than the ASTRAL score in predicting functional outcomes for stroke patients. A study reported an AUC of 0.888 for the neural network model, compared to 0.839 for the ASTRAL score ($P < 0.001$), further highlighting the growing role of ML in the prognosis of stroke.

Research by Wang et al. (2019) demonstrated the precision of Random Forest algorithms in predicting functional outcomes after intracerebral haemorrhage, with AUCs of 0.899 at one month and 0.917 at six months [10]. Lastly, linear SVM regression has been used to predict rehabilitation outcomes in stroke patients, showing promising results for motor and cognitive functions, although it remains challenging to predict the Barthel index. These studies collectively highlight the transformative potential of machine learning to improve stroke outcome predictions and improve clinical decision making.

Abujaber et al. (2023) predict a 90-day prognosis for patients with stroke: a machine learning approach [11]. Stroke remains an important global health issue, ranking as the second leading cause of death worldwide. This study aimed to create and evaluate a machine learning tool to predict the 90-day prognosis of stroke patients after discharge, using the modified Rankin score as the outcome measure. The research analysed data from a national multiethnic stroke registry, which included 15,859 patients with ischemic or hemorrhagic stroke, of whom 7,452 met the study criteria. Feature selection was carried out using correlation and permutation importance methods, and six classifiers were applied, including Random Forest (RF), Classification and Regression Tree, Linear Discriminant Analysis, Support Vector Machine, and K-Nearest Neighbours. The RF model achieved the best performance, with an accuracy of 0.823 and an AUC of 0.893, demonstrating excellent discrimination power. The most influential predictors were stroke type, hospital-acquired infections, admission location, and length of stay. Although the RF model shows potential to tailor patient care and improve stroke prevention, further prospective validation is needed to verify its effectiveness in real-world clinical settings [11].

Research by Rahman et al. (2023) predicts brain stroke using machine learning algorithms and deep neural network techniques. The study aimed to predict the likelihood of an early-stage stroke using both machine learning and deep learning techniques, using a reliable data set for stroke prediction. Various machine learning models, including XGBoost, AdaBoost, LightGBM, Random Forest, Decision Tree, Logistic Regression, K-Nearest Neighbours (KNN), SVM (Linear Kernel), and Naive Bayes, were applied alongside deep learning models, specifically a three-layer and a four-layer artificial neural network (ANN). The Random Forest classifier achieved the highest classification accuracy at 99%, outperforming all other machine learning models. Among deep learning models, the 4-layer ANN showed the best performance with an accuracy of 92.39%, surpassing the 3-layer ANN. In general, the findings revealed that machine learning techniques, particularly Random Forest, outperformed deep neural network models in predicting stroke occurrence [12].

Zhang (2023) research is from radical stroke prediction based on SVM which demonstrates the best overall performance with an accuracy of 0.792, a precision of 0.712, and a high recall of 0.912, resulting in an F1 score of 0.8. This indicates that SVM is particularly effective in identifying true stroke cases (high recall) while maintaining balanced precision and F1 score. Naïve Bayes also performs well, with an accuracy of 0.768, precision of 0.733, and recall of 0.772, and it shows a slightly higher F1 score of 0.7521 compared to other models. Random Forest achieves an accuracy of 0.76, and although its recall (0.842) is relatively high, its precision is lower (0.696), leading to an F1 score of 0.762. KNN matches Random Forest in accuracy (0.76) and shows balanced precision (0.737) and recall (0.736), with an F1 score of 0.737. Logistic regression performs moderately, with an accuracy of 0.752 and an F1 score of 0.739. Lastly, the Decision Tree has the lowest performance, with an accuracy of 0.728 and an F1 score of 0.721. In general, SVM stands out as the most effective model for stroke prediction, especially in recall, which is crucial to accurately identifying stroke cases [13].

SVM stands out as the best performing model for stroke prediction, with a high accuracy of 0.792 and an F1 score of 0.8. Its recall of 0.912 makes it particularly effective in identifying actual stroke cases, which is crucial for minimising missed diagnoses. SVM also balances well with a precision of 0.712, outperforming other models and making it the most reliable choice for real-life stroke prediction applications.

Wu & Fang (2020) research is stroke prediction with Machine Learning Methods among Older Chinese. In this study, the significant class imbalance between stroke and non-stroke cases (approximately 1:19) required the use of data balancing techniques. Regularised Logistic Regression (RLR), Support Vector Machine (SVM), and Random Forest (RF) models were applied to both the original imbalanced data and three balanced datasets generated using Random Over-Sampling (ROS), Random Under-Sampling (RUS), and the Synthetic Minority Over-Sampling Technique (SMOTE). The imbalanced data set yielded high accuracy but extremely low sensitivity (close to 0.00) and an AUC of around 0.50. However, once data balancing techniques were applied, the model performance improved significantly. Although precision and specificity experienced a

slight decrease, sensitivity increased to 0.78 and AUC rose to 0.72, leading to a more balanced and effective model for stroke prediction. For the unbalanced data set, RF had a higher AUC (0.52, 95% CI 0.51–0.53) compared to RLR ($p < 0.01$), though RF's overall performance remained relatively low. No significant differences were observed between SVM and RLR ($p > 0.05$) [14].

In the ROS-balanced dataset, SVM outperformed RLR significantly, achieving an AUC of 0.71 (95% CI 0.68–0.74), while the difference between RF and RLR was not statistically significant ($p > 0.05$). In both the RUS-balanced and SMOTE-balanced datasets, SVM and RF performed similarly to RLR, with no significant differences noted ($p > 0.05$). Overall, data balancing improved the performance of SVM and RF, with SVM showing the greatest AUC improvement in the ROS-balanced dataset [14].

Bandi et al. (2020) research predicts the severity of brain stroke using machine learning. The Random Forest algorithm outperformed other models with an accuracy of 94.23%, a sensitivity of 92.16%, and a specificity of 95.07%, along with a low error rate of 0.04%. Due to its superior performance, it was chosen for further development in the prediction algorithm to improve accuracy. Other models, such as Decision Tree and AdaBoost, also performed well, but Random Forest proved to be the most reliable for this task [15].

The improvised Random Forest model shows significant improvements over the basic version, as indicated in the table. It achieves a high accuracy of 96.97%, precision of 94.56%, sensitivity of 94.9%, specificity of 97.81%, and an F1 score of 94.73%, with a lower error rate of 0.03%. This enhanced model has been integrated into the SPN algorithm to classify stroke risk levels into three categories: low, moderate, and high. Additionally, the SPN algorithm utilises the proposed model to identify key features within the data set that are essential for detecting various types of strokes, including ischemic, intracerebral, and subarachnoid hemorrhagic strokes. These crucial attributes contribute to a better prediction of stroke risk and help to more effectively manage patients [15].

Azam et al. (2020) study focuses on evaluating the efficacy of high-performance machine learning algorithms for the prediction of stroke risk. Three techniques are investigated: Random Forest (RF), Decision Tree (DT) and Logistic Regression (LR). The research also examines important risk variables that lead to stroke, comparing models with and without the smoking status variable. Data preprocessing techniques are used, particularly to balance the data set and improve model performance. With a focus on supporting efforts to prevent strokes, the objective is to compare how well different algorithms predict stroke risk and pinpoint important characteristics that affect the occurrence of strokes. The data set used for the investigation contains 62001 rows and a total of 12 columns. The first eleven columns contain the features that they will use subsequently to forecast the final column, "target(stroke)," which will indicate whether or not the patient will experience a stroke. The 62001 rows are the patient data that we were able to locate in the data set. The results indicate that the RF classifier achieved the highest accuracy at 99.98%, closely followed by the DT classifier with smoking status at 98.78%, while the DT classifier without smoking status recorded an even higher accuracy at 99.46%. In contrast, both LR classifiers performed significantly lower, with accuracies of 71.21% and 81.34%, respectively. These findings highlight the superior predictive capacity of DT and RF models, particularly when incorporating smoking status as a feature, underscoring the importance of feature selection in improving stroke risk prediction [16].

Islam et al. (2021) suggested employing a machine learning algorithm to analyse stroke risk using a healthcare dataset that included a variety of risk factors. There are 5110 observations with 12 attributes in the data set used. Characteristics include age, gender, heart disease, hypertension, type of work, type of residence, average blood sugar level, BMI, smoking status, and stroke. Other factors are independent and stroke is a dependent variable. The algorithms used are Logistic Regression, Decision Tree Classification, K-Nearest Neighbours and Random Forest. They have employed exploratory data analysis (EDA), which helps find patterns, spot anomalies, and create hypotheses, by analysing data sets and summarizing their essential features using data visualisation tools. Feature engineering is used after EDA to convert unprocessed data into useful features that improve model accuracy, especially when working with unbalanced datasets. Synthetic Minority Oversampling Technique (SMOTE) was specifically used in this work to resolve the imbalance in the target variable, which comprises 4908 patients who did not experience a stroke and 201 patients who experienced one stroke. The Random Forest model achieved the highest accuracy on all metrics, recording a precision, recall, and F1 score of 96%. Following in performance, the Decision Tree Classifier (DTC) secured 93% for each metric, while the K-Nearest Neighbours (K-NN) model achieved 90% for both Precision and Recall, with an F1-Score of 90%. Logistic regression performed the least, with a consistent accuracy of 87% across all three metrics. These results indicate the superior efficacy of the Random Forest model in predicting stroke risk compared to other algorithms tested [17].

Alruily et al. (2023) forecast cerebral stroke illnesses; they presented a tuning ensemble RXLM made up of XGBoost, LightGBM, and RF. They used an open-access stroke dataset to predict cerebral stroke risk using Random Forest (RF), Extreme Gradient Boosting (XGBoost) and LightGBM. The dataset was pre-processed using the KNN imputer to handle missing data, feature normalisation, one-hot encoding, and outlier removal. Synthetic Minority Oversampling (SMO) was used to balance the stroke and non-stroke samples after data splitting. Hyperparameter tuning was performed using a random search technique, and the optimal parameters were then integrated into an ensemble model known as RXLM. When this adjusted set of classes was compared with conventional classifiers, all algorithms performed admirably. The data set comprises 249 stroke patients and 4861 normal patients, together with 2994 females, 2115 males, and one other. A training set of 4088 rows (3901, no-stroke and 187 strokes) (80%) and a test set of 1022 rows (20%) were created from the data set [18].

In the first experiment, the performance of four models—Random Forest (RF), XGBoost, LightGBM, and the ensemble RXLM—was evaluated before hyperparameter optimisation.

RXLM achieved the highest scores for precision (96.08%), precision (96.65%), F1 score (96.06%), Kappa (92.16%), and MCC (92.2%), while XGBoost had the highest AUC (99.13%). RF had the highest recall (96.56%), but the lowest scores for precision (94.49%), AUC (98.84%), precision (92.73%), F1 score (94.6%), Kappa (88.98%) and MCC (89.06%). LightGBM performed similarly to RF and XGBoost, with an accuracy of 94.82%. Overall, RXLM outperformed the other models in most metrics except recall, where it had the lowest value at 95.5%.

In the second experiment, the performance of Random Forest (RF), XGBoost, LightGBM, and the ensemble RXLM was evaluated after hyperparameter tuning using the random search technique. The ensemble RXLM outperformed the others, achieving accuracy (96.34%), AUC (99.38%), precision (96.55%), F1-score (96.33%), Kappa (92.68%), and MCC (92.69%). XGBoost had the highest recall at 98.65%, while its other metrics included accuracy (92.42%), AUC (99.12%), precision (87.74%), F1 score (92.87%), Kappa (84.84%), and MCC (85.52%). LightGBM also performed well, with an accuracy of 95.18% and other metrics showing solid results (AUC: 98.86%, recall: 94.91%, precision: 95.45%, F1-score: 95.17%, Kappa: 90.37%, MCC: 90.39%). On the contrary, RF had the lowest performance across all metrics, with accuracy at 84.73%, AUC at 92.75%, and other scores significantly lower than those of the ensemble and the boosted models.

C. Summary of Reviews

This section summarises existing reviews on stroke predictions by categorising them into different domains, which are healthcare, dental, cloud-based software engineering, renewable energy systems, face recognition, sentiment analysis, and plantation, as shown in Table I.

TABLE I. CATEGORISATION OF PREVIOUS STUDIES ACCORDING TO FOUR DIFFERENT DOMAINS

Criteria	Domain	Reference
Random Forest (RF)	A, B, E	[19], [20], [21], [22]
Support Vector Machine (SVM)	A, F, G, H, I	[19], [23], [24], [25]

Note: A-healthcare technology; B-dental; C-cloud-based software engineering; D-renewable energy systems; E-face recognition; F-sentiment analysis; G-plantation.

III. MATERIALS AND METHODS

Fig. 3 shows the research methodology used in this study. The following sub-sections will explain each step in the research.

A. Data Collection

For the stroke prediction project, we collect data from Kaggle, which contains over 5,000 records, each with 12 essential features for analysis. These features include a unique patient ID, gender, age, and binary indicators for hypertension and heart disease. Additionally, it encompasses marital status, work type, residence type, average glucose level, and body mass index (BMI).

B. Data Pre-Processing

Data preprocessing is essential to prepare the data set for stroke prediction and involves several key steps. First, missing values are identified and imputed using appropriate methods, such as mean or mode, or records may be removed if necessary. Techniques for noise reduction are used to improve data quality, including removing duplicates, cleaning data, and applying data transformation, to enhance data quality. Outlier detection is performed using visualisations and box plot methods, followed by capping or removing extreme values to prevent distortion in analysis. Categorical variables are converted into numerical formats through label encoding or one-hot encoding; for example, the gender variable, which includes "female" and "male," is transformed into 0 for female and 1 for male. Feature scaling is carried out using normalisation or standardization to ensure that all features contribute equally to the model. Finally, strategies to address class imbalance, such as oversampling, under sampling, or adjusting class weights, are implemented to improve model performance and ensure that all classes are adequately represented.

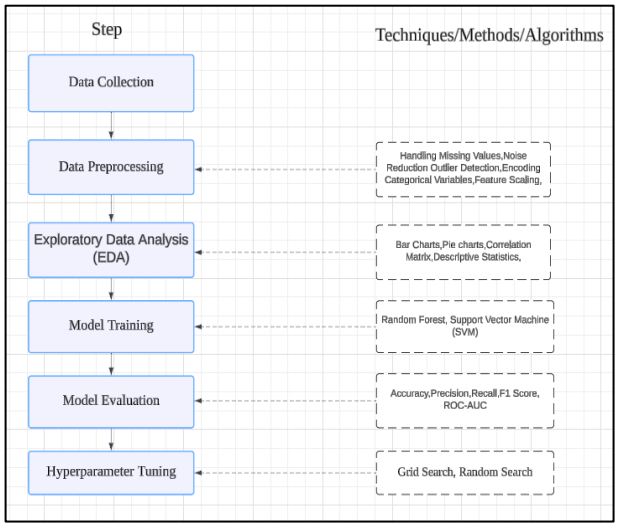


Fig. 3. Research methodology flow diagram.

C. Exploratory Data Analysis (EDA)

Bar charts display the frequency of categorical variables, helping to assess the demographic composition of the data. Pie charts can illustrate the proportions of critical health indicators, such as hypertension and heart disease status. A correlation matrix assesses relationships between numerical features, revealing potential multicollinearity, while descriptive statistics summarise key characteristics, such as mean and standard deviation, aiding in understanding the data's distribution.

D. Model Training

In the model training phase for the stroke prediction project, I will implement and evaluate multiple algorithms, including the Support Vector Machine (SVM) and Random Forest. Specifically, I will explore hybrid approaches that combine SVM with Random Forests to leverage their complementary strengths. The data set will be divided into 80% for training the models and 20% for testing their performance.

E. Model Evaluation

The models in this research are evaluated using Table II equations and evaluation metrics.

TABLE II. MODEL EVALUATION EQUATIONS

Evaluation metrics	Equation/Description
Accuracy	$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Population} = \frac{TP + TN}{TP + TN + FP + FN}$ Note: TP-True Positive; TN-True Negative; TP-Total Population; FP-False Positive; FN-False Negative
Precision	$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} = \frac{TP}{TP + FP}$ Note: TP-True Positive; TN-True Negative; TP-Total Population; FP-False Positive; FN-False Negative
Recall (Sensitivity)	$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} = \frac{TP}{TP + FN}$ Note: TP-True Positive; TN-True Negative; TP-Total Population; FP-False Positive; FN-False Negative
F1 Score	$F1\ Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision + Recall} = 2\ x\ \frac{TP}{2TP + FP + FN}$ Note: TP-True Positive; TN-True Negative; TP-Total Population; FP-False Positive; FN-False Negative
ROC-AUC	Receiver Operating Characteristics - Area under Curve The ROC curve illustrates the trade-off between sensitivity (true positive rate) and specificity (true negative rate) at various threshold settings.
PR-AUC	Precision-Recall Area Under the Curve The purpose of PR-AUC is to evaluate how well a binary classification model identifies the positive class, especially in cases where the data set is imbalanced.

F. Hyperparameter Tuning

Hyperparameter tuning will be performed to enhance model performance, focussing on key hyperparameters, such as learning rate, batch size, number of epochs, and dropout rate. By systematically adjusting these parameters, we aim to optimise the models for stroke prediction and improve their accuracy and generalisation.

IV. RESULTS

A. Dataset

The stroke prediction data set comprises 5,110 observations, each with 12 attributes. Among these, 10 attributes are considered relevant for prediction. These attributes include essential patient details, such as identification number, age, gender, hypertension, marital status, occupation, type of residence, heart disease status, average glucose level, BMI, smoking habits, and stroke status. The data['stroke'].value_counts() output shows the distribution of the target variable stroke, where 4745 customers (92.9%) did not experience a stroke (0), while 365 customers (7.1%) had a stroke (1). The code applies the ggplot style and creates a bar chart to visualise the distribution of stroke cases. It uses Seaborn's countplot with a yellow-green colour palette to show how many customers had a stroke (1) versus those who did not (0). The bar labels display exact counts, making the imbalance in the data set clear: most of the customers did not experience a stroke.

B. Data Pre-Processing

1) Missing values: The data.isnull().sum() output shows the number of missing values in each column (Fig. 4). All columns have complete data except for BMI, which has 184 missing values. This means that of 5110 entries, only 4926 rows have a valid BMI value. The code calculates the percentage of missing values for each column and visualises them using a point plot. First, it creates a DataFrame (missing) with missing value percentages and then plots these values using Seaborn's pointplot. The x-axis represents column names (rotated for readability), and the y-axis shows the percentage of missing data. The BMI column is expected to show around 3.6% missing values (184 out of 5110). This helps identify missing data patterns, guiding decisions on imputation or removal strategies. Only the BMI column has missing values, accounting for 3.6% of the data set. Next, the fillna() code is used to fill in missing values. The inplace=True parameter updates the DataFrame directly. After replacement, it checks for any remaining missing values using isnull().sum(), confirming that all missing BMI values are now replaced (0 missing values remain). This technique helps maintain data consistency without losing records.

2) Noise reduction

a) Duplicates: The data.duplicated() code is used to check for duplicate rows in the dataset. It then filters the data set to display any duplicate rows. The output shows an empty DataFrame, which means that no duplicate rows were found.

b) Distribution and outliers: The boxplot is used to visualise the distribution and outliers for the numerical columns age, BMI, and avg_glucose_level, as shown in Fig. 5. The age column appears to have a well-distributed range from 0 to 80 years, with a median around 45 years and only a few mild outliers. The BMI column shows several outliers above 50, indicating that some individuals have significantly higher BMI values, although most values fall between 10 and 40. The avg_glucose_level column has a large number of extreme outliers, especially above 150 mg/dL, suggesting that some individuals have unusually high glucose levels. This analysis helps identify potential anomalies, which may need to be addressed through techniques such as capping, transformation, or removal to improve data quality for model training.

```
missing = pd.DataFrame((data.isnull().sum()) * 100 / len(data)).reset_index()  # Calculate percentage of missing values

plt.figure(figsize=(16, 5))  # Corrected syntax for figsize
ax = sns.pointplot(x='index', y=0, data=missing)  # Corrected syntax for specifying x and y parameters
plt.xticks(rotation=90, fontsize=7)
plt.title("Percentage of Missing Value")
plt.ylabel("PERCENTAGE")
plt.show()
```

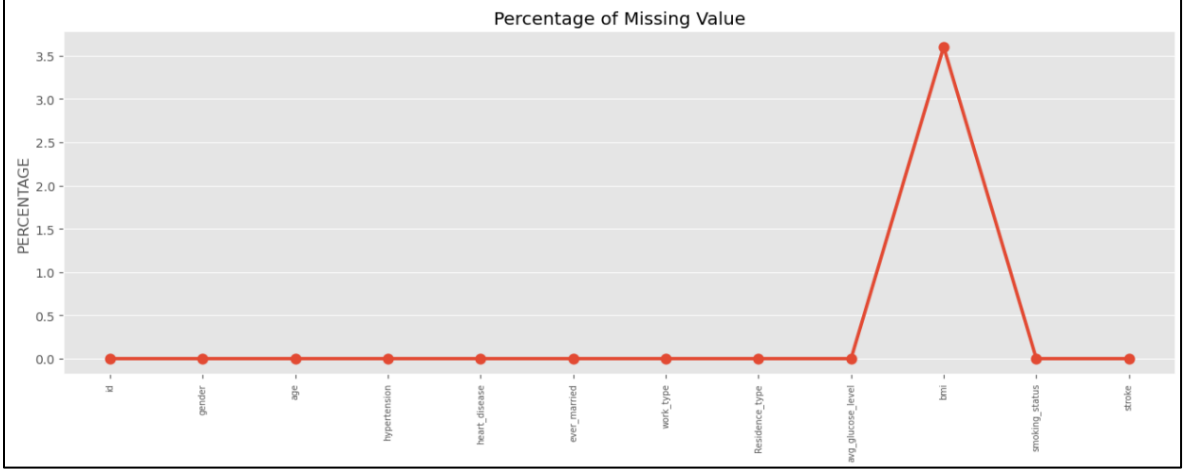


Fig. 4. Data missing analysis and preprocessing.

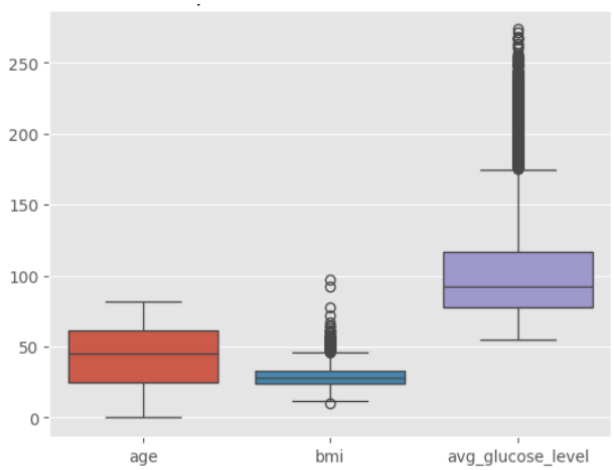


Fig. 5. Box plot for numerical columns.

Data.drop (columns = [‘id’]) code is used to remove the id column from the data set using. The id column is usually a unique identifier for each row and does not contribute to predictive modelling. The removal of it helps reduce the dimensionality and prevents the model from considering it as a meaningful feature, ensuring a cleaner data set for analysis and machine learning.

c) Out-of-range values: The code in Fig. 6 checks for unrealistic values in the columns of age, BMI, and avg_glucose_level by filtering for values outside reasonable ranges (age: 0-82, BMI: 10-55, avg_glucose_level: 0-400). The results show no invalid values for age and glucose level, but some records have BMI values greater than 55, with a maximum of 97.6. These high BMI values may be outliers or data entry errors that require further investigation or treatment (e.g. capping or removal). Identifying and handling such anomalies is crucial to improving model performance and avoiding biased predictions.

Next, the codes in Fig. 7 are used to filter out unrealistic values in the age, BMI, and avg_glucose_level columns by keeping only records within reasonable ranges (age: 0-82, BMI: 10-55, avg_glucose_level: 0-400). It then ensures that any remaining invalid values (although none should exist after filtering) are replaced with the median value of the respective column. This approach removes extreme outliers while maintaining a consistent and realistic, which helps improve model accuracy and reliability.

```
# Check for age outside a reasonable range (0-82)
print(data[(data['age'] < 0) | (data['age'] > 82.0)])

# Check for BMI outside a reasonable range (10-50)
print(data[(data['bmi'] < 10) | (data['bmi'] > 55)])

# Check for avg_glucose_level outside a reasonable range (0-400)
print(data[(data['avg_glucose_level'] < 0) | (data['avg_glucose_level'] > 400)])
```

```
Empty DataFrame
Columns: [gender, age, hypertension, heart_disease, ever_married, work_type, Residence_type, avg_glucose_level, bmi, smoking_status, stroke]
Index: []
      gender   age  hypertension  heart_disease ever_married      work_type  \
1116    Male  63.0             0              0          Yes       Govt_job
1178    Male  62.0             0              0          Yes       Govt_job
1236  Female  61.0             1              0          Yes        Private
1535  Female  57.0             1              0          Yes        Private
1658  Female  56.0             0              0          Yes       Govt_job
1887  Female  53.0             1              0          Yes        Private
```

Fig. 6. Snippet of code used to check for out-of-range values.

```
[38]: data = data[(data['age'] >= 0) & (data['age'] <= 82.0)]
      data = data[(data['bmi'] >= 10) & (data['bmi'] <= 55)]
      data = data[(data['avg_glucose_level'] >= 0) & (data['avg_glucose_level'] <= 400)]

[41]: data.loc[(data['age'] < 0) | (data['age'] > 82.0), 'age'] = data['age'].median()
      data.loc[(data['bmi'] < 10) | (data['bmi'] > 55), 'bmi'] = data['bmi'].median()
      data.loc[(data['avg_glucose_level'] < 0) | (data['avg_glucose_level'] > 400), 'avg_glucose_level'] = data['avg_glucose_level'].median()
```

Fig. 7. Snippet of code used to filter unrealistic values.

```
ranges = {
    'age': (0, 82),
    'bmi': (10, 55),
    'avg_glucose_level': (0, 400)
}

for column, (min_val, max_val) in ranges.items():
    out_of_range = data[(data[column] < min_val) | (data[column] > max_val)]
    print(f"{column}: {out_of_range.shape[0]} out-of-range values")
```

```
age: 0 out-of-range values
bmi: 0 out-of-range values
avg_glucose_level: 0 out-of-range values
```

Fig. 8. Snippet of codes used to define acceptable value ranges.

Fig. 8 codes are used to define acceptable value ranges for the columns age (0-82), BMI (10-55), and avg_glucose_level (0-400) using a dictionary. It then iterates over these columns, filtering the data set to count the number of values that fall outside the specified ranges. The output confirms that no out-of-range values remain in the dataset, which means that all extreme or unrealistic values have been successfully removed or corrected. This ensures a clean data set for analysis and modelling, reducing the risk of biases caused by erroneous data points.

d) Inconsistent categorical values: The code in Fig. 9(a) is used to check for unique values in the categorical column smoking_status to identify inconsistencies. It then standardises values by converting them to lowercase, ensuring uniformity (e.g. changing Unknown to “unknown”). This transformation prevents issues such as case-sensitive mismatches during analysis or modelling. After standardisation, the data set contains four consistent categories: “unknown”, “never smoked”, “formerly smoked”, and “smokes”, improving the data quality and consistency for machine learning or statistical analysis. Fig. 9(b) checks for unique values in the gender column and detects three categories: “Male”, “Female” and “Other”. To ensure consistency, convert all values to lowercase, standardising them as “male”, “female”, and “other”. This step helps avoid case-sensitive mismatches, ensuring that the data remains clean and uniform for analysis and modelling. However, the presence of “other” could require further review, depending on the context and application. Fig. 9(c) checks the unique values in the work_type column and finds five categories: “Private”, “Self-employed”, “Govt_job”, “Never_worked” and “children”. To ensure consistency, it converts all values to lowercase, standardising them as “private”, “self-employed”, “govt_job”, “never_worked” and “children”. This step prevents case-sensitive discrepancies, ensuring that categorical values remain uniform for analysis and modelling. However, further review may be needed to confirm whether “children” and “never_worked” require special handling.

Fig. 9(d) code checks for unique values in the Residence_type column and finds two categories: “Rural” and “Urban”. To ensure consistency, it converts all values to lowercase, standardising them as “rural” and “urban”. This prevents case-sensitive discrepancies that could cause problems in analysis or machine learning models. The final result confirms that all values are now uniform, improving quality and reliability. Lastly, Fig. 9(e) code checks for inconsistencies in the ever_married column by identifying cases where a person is marked as “Yes” (married) but is younger than 18 years old. If such inconsistencies exist, the code corrects them by changing “ever_married” to “No”. However, since no such cases were found in the dataset, no corrections were needed, and the message "No inconsistencies found" was displayed. This ensures that the data remain logical and accurate for analysis.

a) Encoding: The mapping in Fig. 10 represents how categorical variables have been encoded into numerical values for machine learning. The gender is labelled as 0 for the female, 1 for the the male, and 2 for other. Ever_married is binary, with 0 for "No" and 1 for "Yes". Work_type is categorised into five groups: 0 for children, 1 for government jobs, 2 for never worked, 3 for private sector jobs, and 4 for self-employed individuals. Residence_type is divided into 0 for rural and 1 for urban. Smoking_status is assigned values based on smoking history: 0 for formerly smoked, 1 for never smoked, 2 for currently smoke and 3 for unknown status. Lastly, stroke is classified as 0 for no stroke and 1 for stroke occurrence. This encoding standardises the dataset, making it compatible with machine learning models while preserving essential categorical distinctions.

```
# Check for unique values in categorical columns
print(data['smoking_status'].unique())

# Standardize categorical values if needed (e.g., all lowercase)
data['smoking_status'] = data['smoking_status'].str.lower()

# Check again after standardization
print(data['smoking_status'].unique())
```

```
['Unknown' 'never smoked' 'formerly smoked' 'smokes']
['unknown' 'never smoked' 'formerly smoked' 'smokes']
```

(a) Smoking_status

```
# Check for unique values in categorical columns
print(data['gender'].unique())

# Ensure consistency in gender (e.g., make all values lowercase)
data['gender'] = data['gender'].str.lower()

# Check if all values are now consistented
print(data['gender'].unique())
```

```
['Male' 'Female' 'Other']
['male' 'female' 'other']
```

(b) Gender

```
# Check for unique values in categorical columns
print(data['work_type'].unique())

# Ensure consistency in work_type (e.g., make all values lowercase)
data['work_type'] = data['work_type'].str.lower()

# Check if all values are now consistented
print(data['work_type'].unique())
```

```
['Private' 'Self-employed' 'Govt_job' 'Never_worked' 'children']
['private' 'self-employed' 'govt_job' 'never_worked' 'children']
```

(c) Work_type

```
# Check for unique values in categorical columns
print(data['Residence_type'].unique())

# Ensure consistency in Residence_type (e.g., make all values lowercase)
data['Residence_type'] = data['Residence_type'].str.lower()

# Check if all values are now consistented
print(data['Residence_type'].unique())
```

```
['Rural' 'Urban']
['rural' 'urban']
```

(d) Residence_type

```
# Check inconsistencies
inconsistent_data = data[(data['ever_married'] == 'Yes') & (data['age'] < 18)]

if not inconsistent_data.empty:
    # Correct 'ever_married' to 'No' for inconsistent rows
    data.loc[(data['ever_married'] == 'Yes') & (data['age'] < 18), 'ever_married'] = 'No'
    print("Inconsistencies corrected by updating 'ever_married' to 'No'.")
else:
    print("No inconsistencies found.")
```

```
No inconsistencies found.
```

(e) Ever_married

Fig. 9. Snippet of code used to check for unique values.

```
from sklearn.preprocessing import LabelEncoder, OrdinalEncoder

# Columns for Ordinal and Label Encoding
ordinal_encode_columns = ['ever_married', 'smoking_status']  # Nominal variables
label_encode_columns = ['Residence_type', 'work_type', 'gender']  # Suitable for Label Encoding

# Verify columns before applying Ordinal Encoding
ordinal_encode_columns = [col for col in ordinal_encode_columns if col in data.columns]
if ordinal_encode_columns:
    ordinal_encoder = OrdinalEncoder()
    data[ordinal_encode_columns] = ordinal_encoder.fit_transform(data[ordinal_encode_columns]).astype(int)
else:
    print(f"No valid columns found for Ordinal Encoding: {ordinal_encode_columns}")

# Verify columns before applying Label Encoding
label_encoder = LabelEncoder()
for column in label_encode_columns:
    if column in data.columns:
        data[column] = label_encoder.fit_transform(data[column])
    else:
        print(f"Column not found for Label Encoding: {column}")

# Display the resulting dataset
print("Data after applying Ordinal and Label Encoding techniques:")
data.head(5)
```

Data after applying Ordinal and Label Encoding techniques:

	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	1	82.0	1	1	1	3	0	208.30	32.5	3	1
1	0	82.0	1	1	1	4	1	196.92	22.2	1	1

Fig. 10. Categorical variables encoded as numerical variables.

C. Exploratory Data Analysis (EDA)

To understand the class distribution of the target variable stroke after data preprocessing, we analyzed both the count and proportion of each class (0 = no stroke, 1 = stroke) in the dataset. After data preprocessing, the count for “No Stroke” is 4713 instances and “Stroke” is 364 instances: with a proportion of 92.83% “No Stroke” and 7.17% “Stroke”. The imbalance is a common issue in medical datasets and will be addressed during model development by using class-weighted algorithms to mitigate this issue and improve the model's ability to detect stroke cases accurately.

1) Univariate Analysis (Distribution of Individual Variables): The bar graph in Fig. 11 shows the frequency distribution of various categorical characteristics in the data set. This graph helps us to understand the composition and balance of each feature. By this bar chart, we can notice that most of the

gender in the dataset is female ‘0’ and the least is the other ‘2’. Most patients in our data set do not have hypertension ‘0’ and heart disease ‘0’ and most are married ‘1’. Most of the patients are under the private company ‘2’, and the least of them are under the ‘2’ category and have never worked. The resident of Urban ‘1’ is slightly higher than Rural ‘0’. Lastly, most patients never smoked ‘1’; there is almost the same proportion of patients who previously smoked ‘0’ and smoke ‘2’.

Meanwhile, the histogram in Fig. 12 displays the frequency distribution of various numerical features in the data set. This graph helps us to understand the composition and balance of each feature. By using the histogram, we can notice that most patients in the data set are between 50 and 60 years old with an appropriate frequency of 260. For the BMI variable, most of the data is around 28 to 30 with appropriate 630 frequencies. Lastly, most average glucose levels are around 80 to 90 with appropriately 660 frequencies.

2) Bivariate Analysis (Relationship between Variables and Stroke): The bar charts in Fig. 13 display the frequency distribution of various categorical features in the dataset. This visualisation helps us to understand the composition and balance of each feature and its potential relationship with stroke occurrence. From the chart, we observe the following:

- Gender: The number of female patients who experienced a stroke is slightly higher than that of male patients.
- Hypertension or heart disease: Patients with hypertension or heart disease appear to be more likely to have had a stroke compared to those without these conditions.
- Marital Status: The number of stroke cases is higher among married patients than among single patients.
- Work Type: Patients employed in the private sector make up the largest proportion of both stroke and non-stroke cases.
- Residence Type: The deaths and non-stroke cases in urban areas are slightly higher than in rural areas.
- Smoking Status: Among patients who never smoked, most did not have a stroke. In contrast, people who currently smoke have the highest number of stroke cases.

Based on these observations, we can make a preliminary assumption that individuals who are married females, with hypertension and heart disease, work in the private sector, and have a habit of smoking, may be at higher risk of stroke.

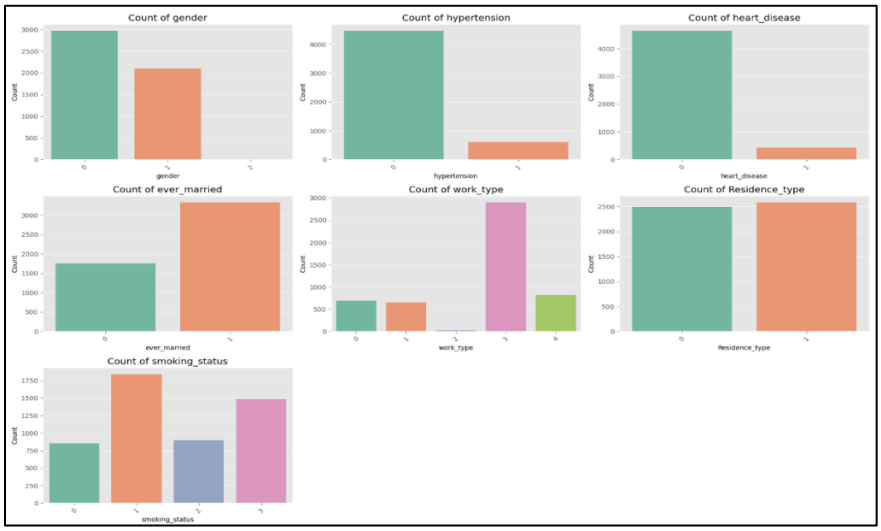


Fig. 11. Frequency distribution of various categorical features.

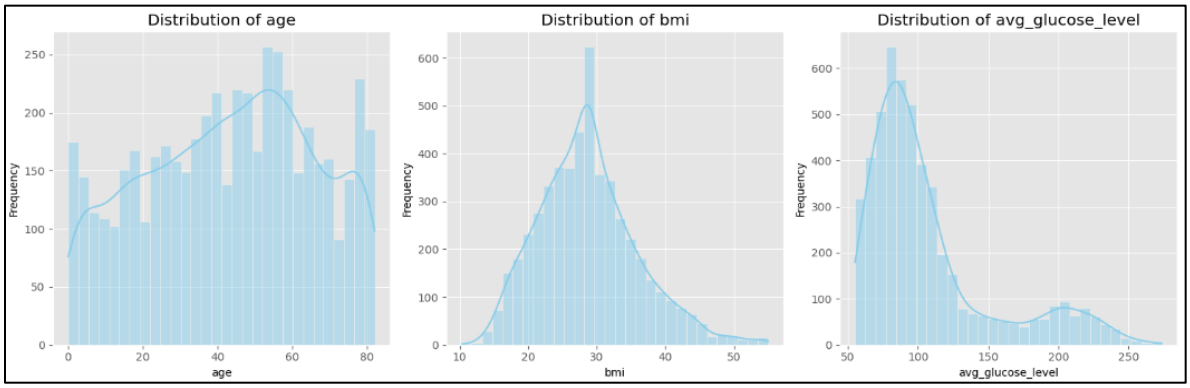


Fig. 12. Frequency distribution of various numerical features.

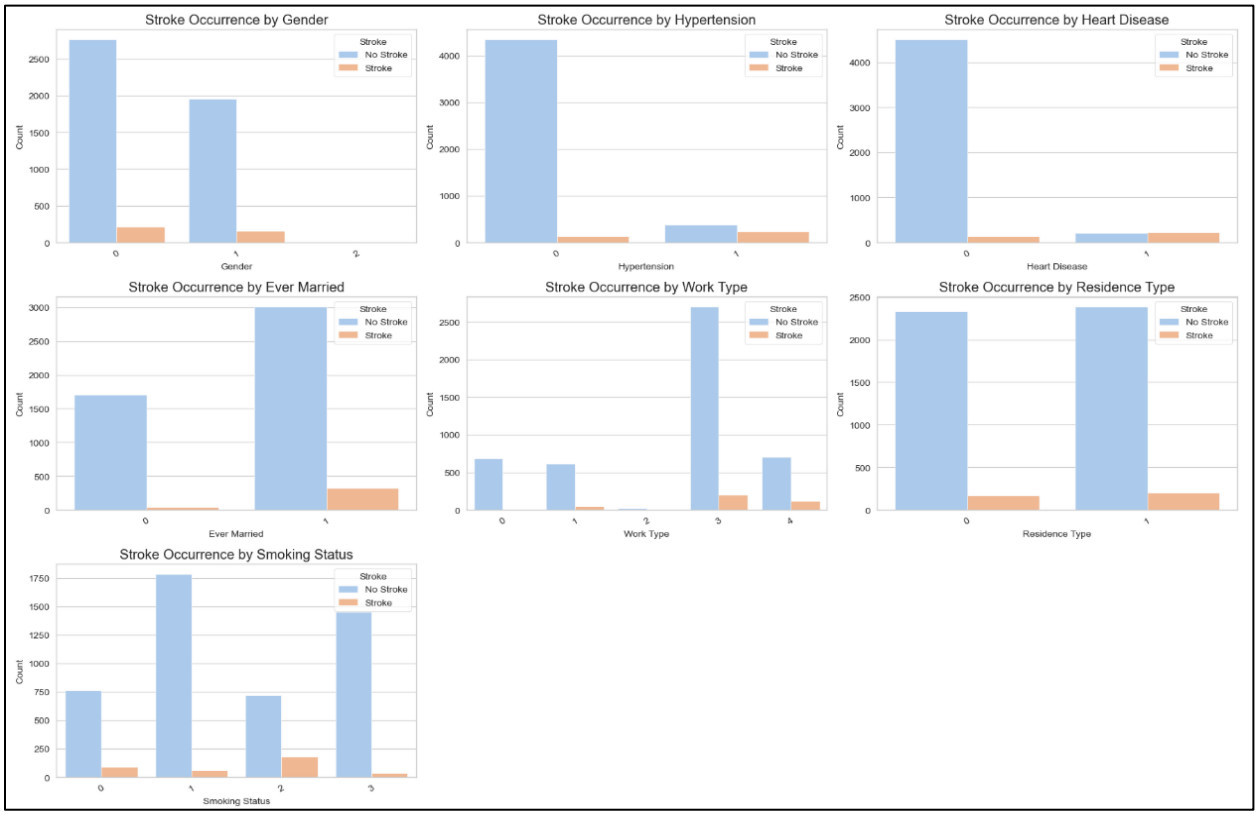


Fig. 13. Frequency distribution of various categorical features in the dataset.

3) Association Between Categorical Variables (Chi-Square Test): To examine whether there is a statistically significant association between categorical variables and stroke occurrence, a Chi-Square Test of Independence was performed for each categorical variable in the data set. According to Table III, variables such as "hypertension", "heart_disease", "ever_married", "work_type", and "smoking_status" show a statistically significant relationship with stroke. This suggests that these features might be useful predictors of stroke risk in further modelling or analysis. The variables "gender" and "residence_type" do not show a statistically significant association with stroke. This implies that these variables may not play a substantial role in determining stroke risk in this data set.

TABLE III. CHI-SQUARE STATISTICS

Variables	Chi-Square Statistics	p-value
gender	0.1907	0.9090
hypertension	1031.71	$2.2995e^{-226}$
heart_disease	1364.37	$1.1635e^{-298}$
ever_married	86.0216	$1.7798e^{-20}$
work_type	113.968	$1.0362e^{-23}$
residence_type	1.5532	0.2127
smoking_status	321.95	$1.7643e^{-69}$

The box plots in Fig. 14 illustrate the distribution of three continuous variables "age", "average', "avg_glucose_evel" and "BMI" between stroke outcomes (0 = no stroke, 1 = stroke). These visualisations help us identify how these variables differ between stroke and non-stroke patients. Stroke patients tend to be significantly older than non-stroke patients. The median age of stroke patients is around 75, while for non-stroke patients it is closer to 40. There is a wider age range among non-stroke patients, whereas stroke cases are more concentrated among the elderly. This suggests that age is a strong risk factor for stroke. Patients who have had a stroke tend to have higher average glucose levels. The median glucose level for stroke patients is above 150, compared to below 100 for non-stroke patients. The presence of high outliers in both groups indicates variability, but stroke patients generally show higher glucose distributions. This supports the idea that high blood sugar or diabetes may be associated with an increased risk of stroke. The distribution of stroke and non-stroke patients shows some overlap, but stroke patients have a slightly higher median BMI. There are more extreme BMI values (outliers) in the non-stroke group. Although the difference is less pronounced compared to age and glucose level, it still suggests a potential moderate relationship between obesity and stroke.

4) Correlation matrix: The correlation matrix in Fig. 15 shows Pearson's correlation coefficients between the numerical and binary variables in the data set. These coefficients range from -1 to 1. We are especially interested in how each feature correlates with the target variable stroke. The features with a stronger association with stroke ($r \geq 0.3$) are the features may have a more significant influence on stroke occurrence. Heart disease ($r = 0.52$) is the strongest positive correlation with stroke. This indicates that people with heart disease are more likely to suffer a stroke. Hypertension ($r = 0.45$) was positively

correlated, which means that high blood pressure can increase the risk of stroke. Meanwhile, age (r = 0.35) shows that older individuals are more prone to stroke. For the average glucose level (r = 0.30), elevated glucose levels are associated with a higher risk of stroke.

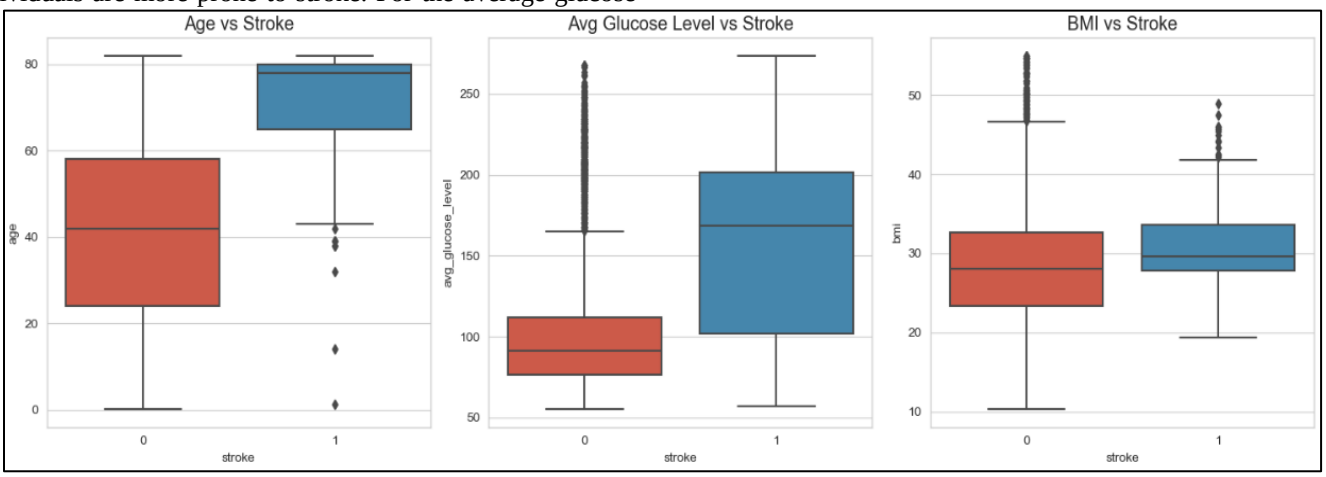


Fig. 14. Distribution of variables.

On the other hand, features with weak or No Significant Correlation to Stroke (r < 0.3) are variables do not show a strong linear relationship with stroke: Gender (r = 0.00), ever married (r = 0.13), type of work (r = 0.12), type of residence (r = 0.02), BMI (r = 0.08) and smoking status (r = -0.04). Although these features might still have some predictive power (especially in nonlinear models), they are not strongly linearly associated with stroke. Thus, through this heat map, we know that "heart_disease", "hypertension", "age", and "avg_glucose_level" are the most relevant features correlated with stroke. Features such as "gender", "ever_married", and "BMI" show little to no correlation, suggesting that they may play a less significant role in stroke prediction from a linear perspective.

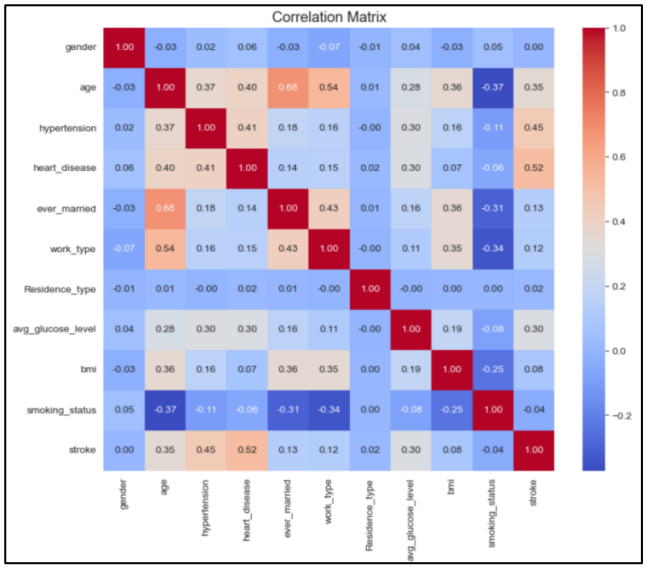


Fig. 15. Correlation matrix of variables.

5) Feature importance: To complement the correlation analysis, a RandomForestClassifier was trained to identify the most influential features in the prediction of stroke (Fig. 16). This model is effective because it captures not only linear relationships, but also non-linear patterns and interactions between features. According to the Random Forest model, the features with the highest scores (greater than 0.05) are age (0.29), avg_glucose_level (0.28), heart_disease (0.13), BMI (0.122), hypertension (0.12), and smoking_status (0.06). These variables contribute significantly to the predictions and are strongly associated with the probability of stroke.

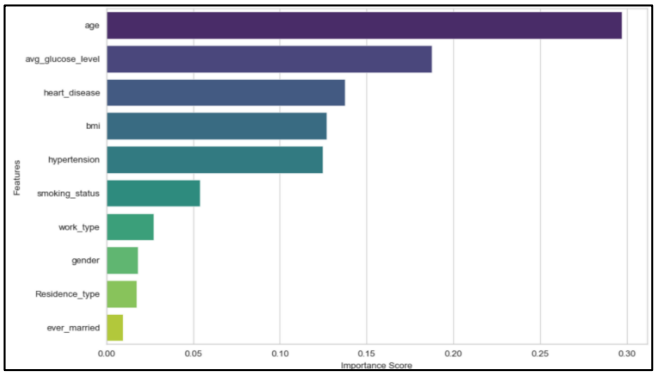


Fig. 16. Feature importance analysis (for EDA).

On the other hand, the features with lower importance scores (less than 0.05) are: work_type (0.025), gender (0.02), residence_type (0.019), ever_married (0.01). These variables may still play minor roles in prediction but are not considered strong indicators of stroke in this model compared to clinical factors such as age, "avg_glucose_level" or "heart_disease".

After reviewing the findings of the Chi-square test analysis, correlation matrix, and the result of the importance of features, we observed that both "gender" and "residence_type" consistently showed a low association with stroke. As a result, we decided to exclude these two variables from the final model to improve processing efficiency and eliminate weak predictors.

D. Split the Data for Training, Testing and Validation

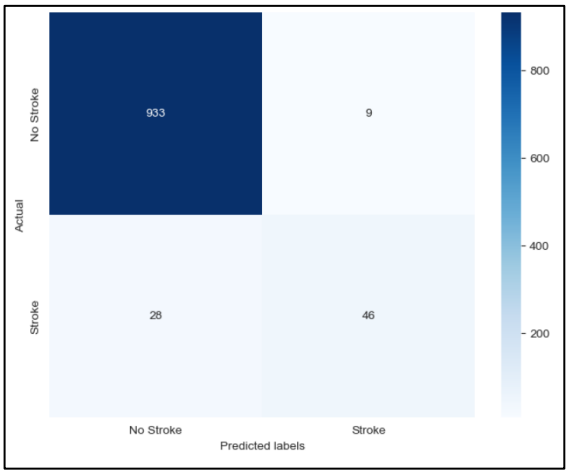
The data set has been pre-processed by separating the target variable (stroke) from the feature set and applying MinMax Scaling to normalise all feature values between 0 and 1. This ensures that all features contribute equally to the model. The scaled data set is then divided into training subsets (70%), validation (10%), and testing (20%) using the split train-test. First, 70% of the data are assigned to training, while the remaining 30% is temporarily stored. Then, this temporary set is further split into 10% for validation and 20% for testing. The final data set sizes are (3553, 10) for training, (508, 10) for validation, and (1016, 10) for testing, ensuring a well-balanced distribution for model training, tuning, and evaluation.

The SVM (Support Vector Machine) classifier is then evaluated on different training subset sizes to analyse its impact on classification performance. It first subsets the training data to specified sizes, applies MinMax Scaling, and then trains the SVM model on the subset. The model is tested in a separate test set and its accuracy, precision, recall, and F1 score are measured. The results show consistent performance across different subset sizes, with accuracy around 96.4%, indicating that the model generalises well with even smaller training data. The purpose of this experiment is to assess how the size of the dataset influences the performance of the model and determine whether a reduced dataset can achieve similar predictive accuracy, optimising computational efficiency.

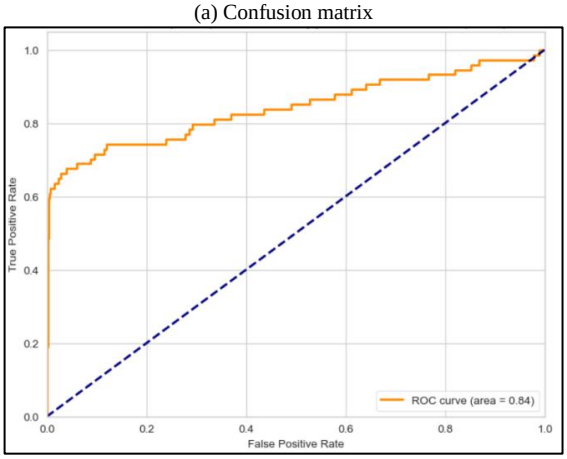
E. SVM

The Support Vector Machine (SVM) model achieved a high accuracy of 96.36%, indicating strong overall performance. The model effectively identified non-stroke cases, with a precision of 97% and a recall of 99%, meaning it rarely misclassified healthy individuals as stroke patients. However, its ability to detect actual stroke cases was moderate, with a precision of 83.64% and a recall of 62.16%, indicating that it correctly identified only 62.16% of actual stroke cases, while misclassifying some as non-stroke. The F1 score of 71.32% reflects a balance between precision and recall for stroke prediction.

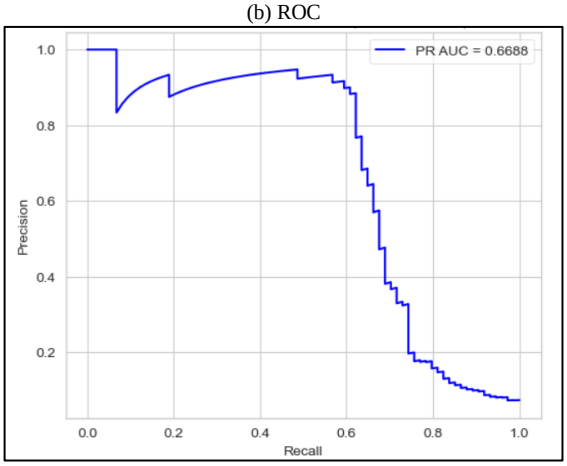
The confusion matrix Fig. 17(a) shows that the model correctly classifies 933 “No Stroke” cases and 46 “Stroke” cases, achieving high overall accuracy (96.36%). However, it misclassifies 28 actual stroke cases as “No Stroke”, leading to a low recall of 62.16%, meaning the model fails to detect nearly 38% of stroke cases, which is critical in medical applications. Although precision for stroke detection is high (83.64%), indicating that most predicted strokes are correct, the model's bias toward the majority class results in missed stroke cases. The Receiver Operating Characteristics (ROC) curve [Fig. 17(b)] for the Support Vector Machine (SVM) model demonstrates its ability to distinguish between stroke and nonstroke cases. The AUC score of 0.8418 indicates that the model has a good level of discrimination, which means that it correctly ranks stroke cases higher than non-stroke cases approximately 84.18% of the time. The Precision-Recall (PR) curve [Fig. 17(c)] evaluates the model performance in distinguishing stroke cases, especially when dealing with an imbalanced dataset. The PR AUC score of 0.6688 indicates that the Support Vector Machine (SVM) model achieves a moderate balance between precision (how many predicted stroke cases are actually strokes) and recall (how many actual stroke cases are correctly identified).



(a) Confusion matrix



(b) ROC



(c) Precision Recall

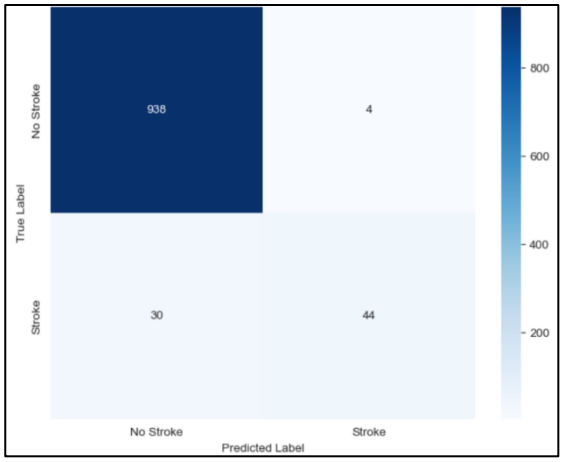
Fig. 17. SVM performance evaluation.

F. Random Forest

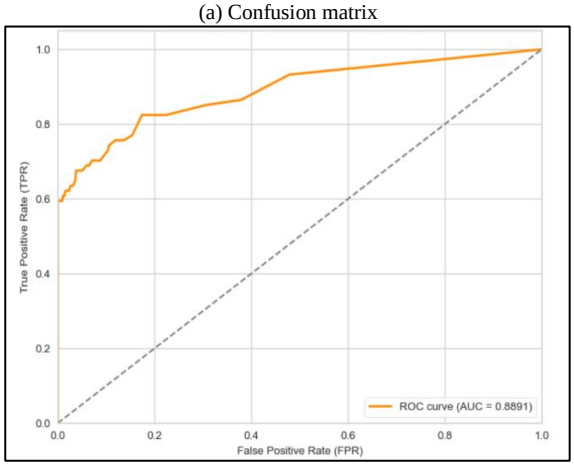
The Random Forest Classifier achieved a high accuracy of 96.75%, indicating a strong overall performance. The precision for stroke cases (1) is 93.62%, which means that most predicted stroke cases were correct. However, the recall is only 59.46%, suggesting that the model struggles to identify all actual stroke cases, leading to a significant number of false negatives. The F1

score of 72.73% reflects this trade-off between precision and recall.

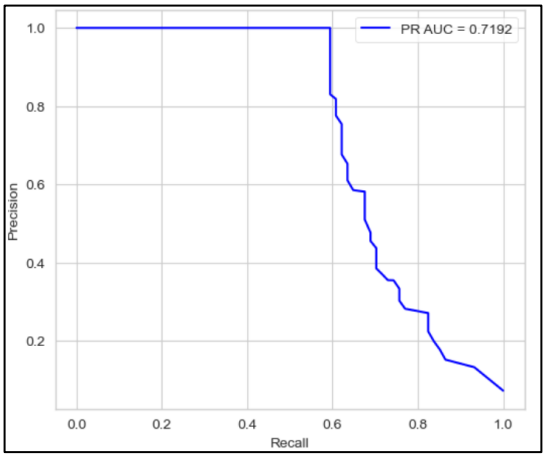
The confusion matrix in Fig. 18(a) shows that the model correctly classifies 938 "No Stroke" cases and 44 "Stroke" cases, achieving high overall accuracy (96.75%). It excels at correctly identifying nonstroke cases (0 class, with near-perfect recall) with only misclassified four non-stroke cases as stroke (False Positives). However, it misclassifies 30 actual stroke cases as "No Stroke", leading to a low recall of 59.46%, meaning that the model fails to detect nearly 41% of stroke cases, which is critical in medical applications. While precision for stroke detection is high (93.62%), indicating that most predicted strokes are correct, the model's bias toward the majority class results in missed stroke cases. The Receiver Operating Characteristics (ROC) curve [Fig 18(b)] for the Random Forest (RF) model demonstrates its ability to distinguish between stroke and nonstroke cases. The AUC score of 0.8891 indicates that the model has a good level of discrimination, meaning it correctly ranks stroke cases higher than non-stroke cases approximately 88.91% of the time. The Precision Recall (PR) curve [Fig. 18(c)] evaluates the performance in distinguishing stroke cases, especially when dealing with an unbalanced data set. The PR AUC score of 0.7192 indicates that the Random Forest (RF) model achieves a moderate balance between precision (how many predicted stroke cases are actually strokes) and recall (how many actual stroke cases are correctly identified).



(a) Confusion matrix



(b) ROC



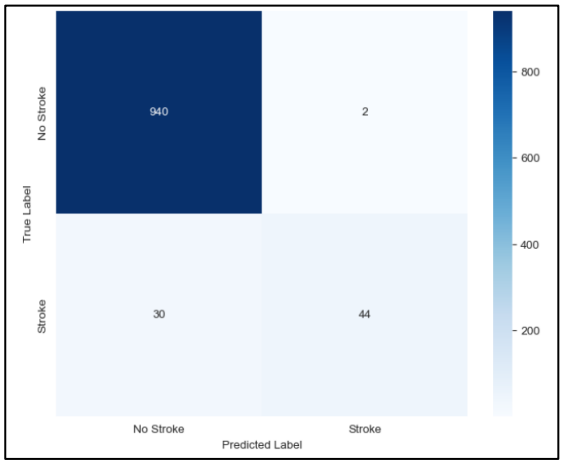
(c) Precision Recall

Fig. 18. Random Forest performance evaluation.

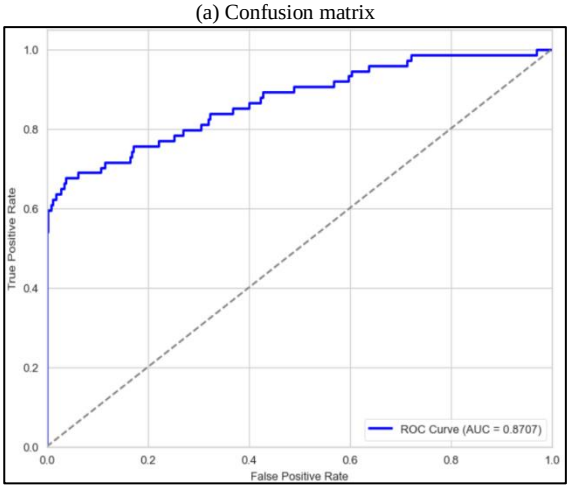
G. Hybrid Feature Augmentation Model (RF+SVM)

The feature-augmented SVM model, enhanced with Random Forest predictions, achieved a high test accuracy of 96.85%, with a precision of 95.65%, recall of 59.46%, and an F1-score of 0.7333. This indicates that the hybrid model is highly effective in correctly identifying stroke cases while minimising false positives. Although the recall is moderate, it reflects a significant improvement in capturing true stroke cases compared to standard models.

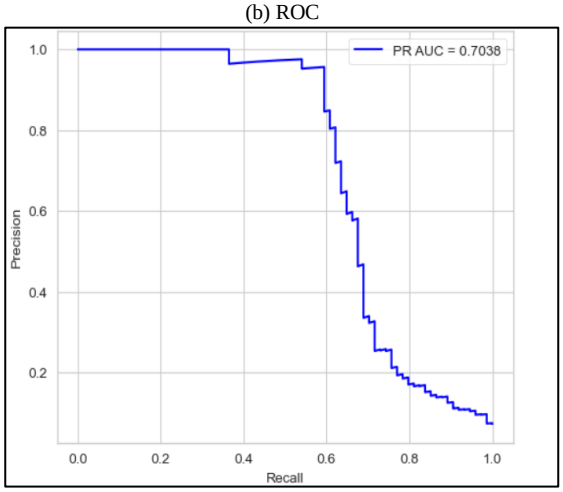
The confusion matrix in Fig. 19(a) shows that the model correctly classifies 940 cases of “No Stroke” and 44 cases of “Stroke”, achieving high overall accuracy (96.85%). It excels at correctly identifying non-stroke cases (0 class, with near-perfect recall), with only misclassifying two non-stroke cases as stroke (False Positives). However, it misclassifies 30 actual stroke cases as "No Stroke", leading to a low recall of 59.46%, meaning that the model fails to detect nearly 41% of stroke cases, which is critical in medical applications. While the precision for stroke detection is high (95.65%), indicating that most predicted strokes are correct, the model's bias toward the majority class results in missed stroke cases. The augmentation model (SVM+RF) in Fig. 19(b) achieves an AUC score of 0.8707, indicating strong classification performance. An AUC close to 1.0 suggests that the model is effective in distinguishing between stroke and non-stroke cases across various threshold levels. This score reflects a good balance between sensitivity (true positive rate) and specificity (1 false positive rate), supporting the reliability in medical decision-making where accurate classification is crucial. It means that it correctly ranks stroke cases higher than non-stroke cases approximately 87.07% of the time. The Precision-Recall (PR) Curve analysis [Fig. 19(c)] for the feature-augmented SVM model yields a PR AUC score of 0.7038, indicating a moderately strong ability to identify stroke cases, especially given the class imbalance. This score reflects how well the model maintains high precision and recall when predicting the minority class (stroke). A PR AUC closer to 1.0 suggests better performance in minimising false positives while capturing most true stroke cases. Thus, the value of 0.7038 demonstrates that the model is reasonably effective in prioritising stroke detection with fewer false alarms.



(a) Confusion matrix



(b) ROC



(c) Precision Recall

Fig. 19. Performance evaluation of the hybrid SVM + RF model.

H. Model Evaluation (RF, SVM, Feature Augmentation)

The bar chart in Fig. 20 compares the Support Vector Machine (SVM), the Random Forest (RF) and a hybrid Augmented SVM model on four performance metrics: accuracy, precision, recall, and F1 score. Although SVM shows the highest recall (0.622), it falls behind in precision and F1 score, indicating that it captures more positives, but also introduces more false alarms. Random Forest performs more consistently, especially in precision (0.917), but still lags slightly behind in recall and overall balance. The feature-augmented SVM, which combines the strengths of both SVM and RF, achieves the best overall performance with the highest accuracy (0.969), precision (0.957), and F1 score (0.733), while maintaining recall comparable to RF. This improvement demonstrates that integrating the high recall ability of SVM with the precision strength of Random Forest leads to a more robust and well-rounded hybrid model.

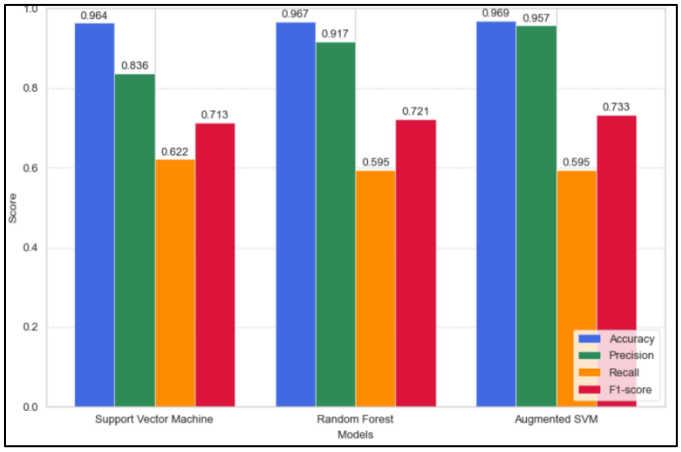


Fig. 20. Performance comparison between Random Forest, Support Vector Machine, and hybrid model (SVM+RF).

The confusion matrices for the SVM, Random Forest (RF), and Feature-Augmented Model (SVM+RF) in Fig. 21 illustrate how each model performs in classifying stroke and non-stroke cases. The SVM model correctly predicts 933 nonstroke cases and 46 stroke cases, while misclassifying 9 non-stroke cases as stroke and 28 stroke cases as non-stroke. Both the RF and Feature-Augmented models correctly predict 44 stroke cases and misclassify 30 stroke cases as non-stroke. However, RF correctly classifies 938 non-stroke cases and misclassifies 4 as stroke, while the Feature-Augmented model improves further by correctly predicting 940 non-stroke cases and misclassifying only 2. This demonstrates that combining the strengths of SVM and RF reduces the number of false positives (non-stroke cases wrongly classified as stroke), enhancing the model's precision by reducing unnecessary stroke predictions, leading to a more reliable and balanced classification system.

The comparison of the Precision Recall (PR) curve in Fig. 19 illustrates the trade-off between precision and recall for the SVM, Random Forest, and Feature-augmented model. Among the three, Random Forest achieves the highest PR AUC score (0.7192), indicating a more stable performance across all thresholds. Although the hybrid model shows superior point metrics (higher precision and F1 score at the default threshold), its slightly lower PR AUC (0.7038) suggests less consistent performance across varying thresholds. Similarly, the SVM has a PR AUC of 0.7101, reflecting moderate performance. In general, the hybrid model demonstrates strong effectiveness at the default classification threshold, while the Random Forest offers more reliable confidence across a range of threshold values.

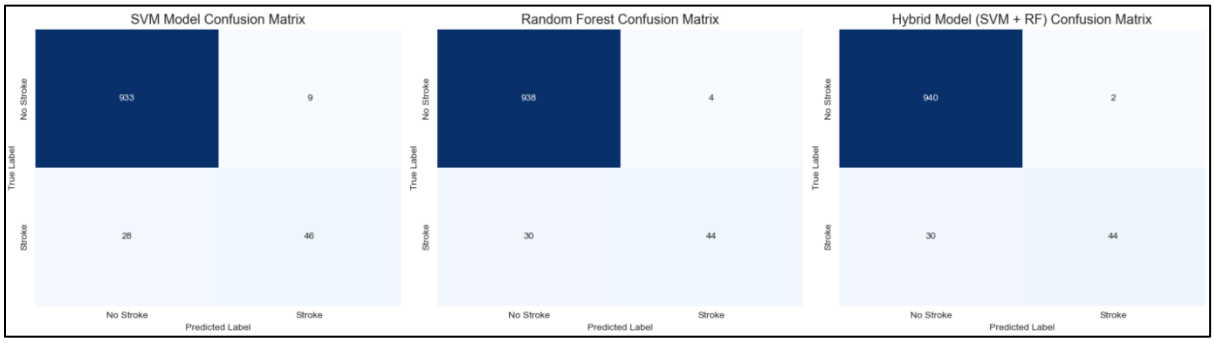
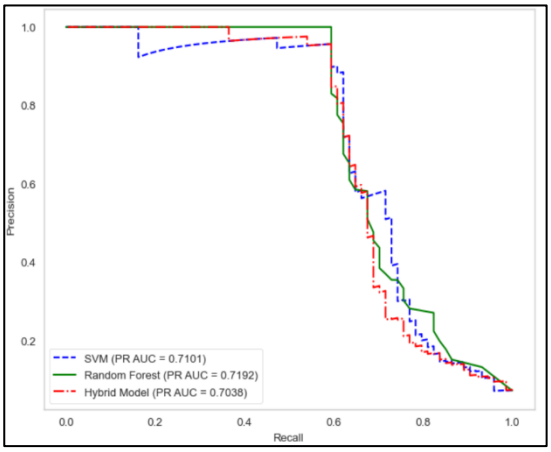
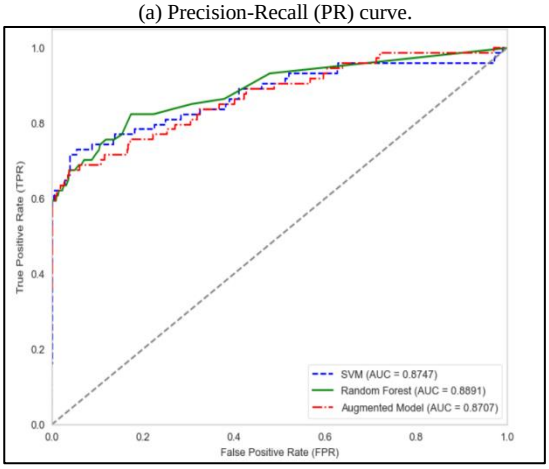


Fig. 21. Confusion matrices for the SVM, Random Forest (RF), and Feature-Augmented Model (SVM+RF).



(a) Precision-Recall (PR) curve.



(b) ROC AUC scores.

Fig. 22. Performance comparison for the SVM, Random Forest (RF) and hybrid model (SVM+RF).

The ROC AUC scores for the models are as follows: the Augmented (Hybrid) model achieved 0.8707, the Random Forest model achieved 0.8891, and the SVM model achieved 0.8747 (Fig. 22). These scores reflect the ability of each model to distinguish between stroke and non-stroke cases, with the Random Forest model showing the highest performance in this regard. The Augmented (Hybrid) model, although slightly lower in the ROC AUC compared to Random Forest, demonstrated improvements in other key evaluation metrics, such as precision and the F1 score, compared to the individual models. This suggests that the hybrid model effectively balances performance across different metrics, making it a viable alternative in stroke classification tasks.

I. Hyperparameter-Tuning Feature Augmented Model (SVM+RF)

1) Grid search: The feature augmented model was developed to improve model performance in predicting stroke cases using a VotingClassifier that combines Support Vector Machine (SVM) and Random Forest (RF) classifiers. Both models were optimised using GridSearchCV with 5-fold cross-validation and class weight adjustments to address class imbalance. The best SVM model used a linear kernel, C = 10, and class_weight = "balanced", while the best RF model used 200 estimators, max_depth = 10, min_samples_split = 5, and class_weight = "balanced_subsample". The final ensemble applied soft voting based on predicted probabilities. An optimal decision threshold (0.5805) was selected using the precision-recall curve to maximise the F1-score. The tuned model achieved an accuracy of 96.85%, a precision of 95.65%, a recall of 59.46%, and an F1 score of 73.33%. Furthermore, the model achieved a PR AUC of 0.7262 and a ROC AUC of 0.9031, indicating strong overall performance and improved ability to identify stroke cases while effectively managing the class imbalance.

2) Random search: Another way to perform hyperparameter tuning is to optimise Random Forest using RandomizedSearchCV with 5-fold cross-validation, and its predicted probabilities and class labels were appended as new features to the original dataset. These augmented data were used to train an SVM, also optimised via RandomizedSearchCV. Evaluated on a test set with an optimised for recall, the hybrid model achieved an accuracy of 97.05%, perfect precision (1.0000), a recall of 59.46%, and an F1 score of 0.7458. Despite slightly lower ROC AUC scores (0.8421) and PR AUC (0.6886) compared to the standalone Random Forest, the tuned hybrid approach significantly improved precision while maintaining reasonable recall, demonstrating its efficacy in balancing high precision with improved detection of stroke cases.

After comparing the results of the two hyperparameter tuning methods, we decided to proceed with the Random Search outcome, as it demonstrated superior classification performance compared to Grid Search. This decision aligns with the objective

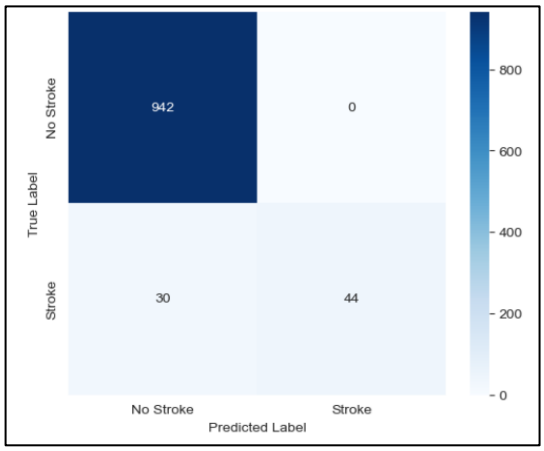
of this investigation, which is to improve the overall performance of the model. The tuned feature augmented model performance is saved for later development usage.

3) Performance evaluation: The confusion matrix in Fig. 23(a) shows that the model correctly classifies 942 "No Stroke" cases and 44 "Stroke" cases, achieving high overall accuracy (97.05%). It excels at correctly identifying non-stroke cases (0 class, with near-perfect recall) without misclassifying any non-stroke cases as stroke (False Positives). However, it misclassifies 30 actual stroke cases as "No Stroke", leading to a low recall of 59.46%, meaning the model fails to detect nearly 41% of stroke cases. Although precision for stroke detection is high (100%), indicating that most predicted strokes are correct, the model's bias toward the majority class results in missed stroke cases. The ROC AUC score for the hybrid model [Fig. 23(b)] after applying random search for hyperparameter tuning is 0.8421, indicating the strong ability to distinguish between stroke and non-stroke cases. The ROC curve rises steeply towards the top left corner, reflecting a high true positive rate (TPR) and a low false positive rate (FPR), suggesting effective classification performance. The Precision-Recall (PR) Curve analysis [Fig. 23(c)] for the feature-augmented SVM model resulted in a PR AUC score of 0.7038, indicating a moderately strong ability to identify stroke cases, particularly in the context of class imbalance. After tuning through Random Search, the model's PR AUC score decreased slightly decreased to 0.6886. This change reflects the model's performance adjustment after the tuning process, aiming for an improved balance between precision and recall.

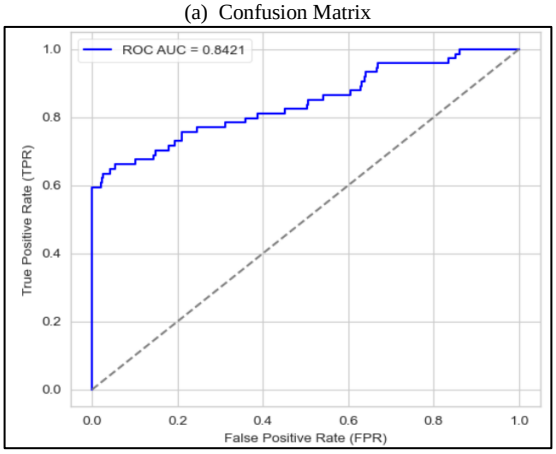
J. *Model Evaluation After Hyperparameter Tuning (SVM, RF, SVM+RF, Tuned SVM+RF)*

The bar chart in Fig. 24 compares the performance of four models: Support Vector Machine (SVM), Random Forest, Feature-Enhanced SVM and Tuned Feature Augmented SVM using accuracy, precision, recall, and F1 score. Among them, the tuned Augmented SVM achieved the highest accuracy (0.970), high precision (1.000), and F1 score (0.746), indicating a strong ability to correctly identify positive cases while balancing precision and recall. Although its recall (0.595) matches that of Random Forest and Feature Augmented SVM, its exceptional precision significantly increases its overall effectiveness. Random Forest showed high accuracy (0.967) and strong precision (0.917), but a lower recall (0.595) reduced its F1 score to 0.721. The standard SVM performed well with high accuracy (0.964) and better recall (0.622), resulting in an F1 score of 0.713. The enhanced SVM improved over the standard SVM with higher precision (0.957) and an F1 score (0.733), although its recall remained unchanged. Overall, the Tuned Augmented SVM demonstrated the best balance between metrics, highlighting the impact of augmentation and hyperparameter tuning.

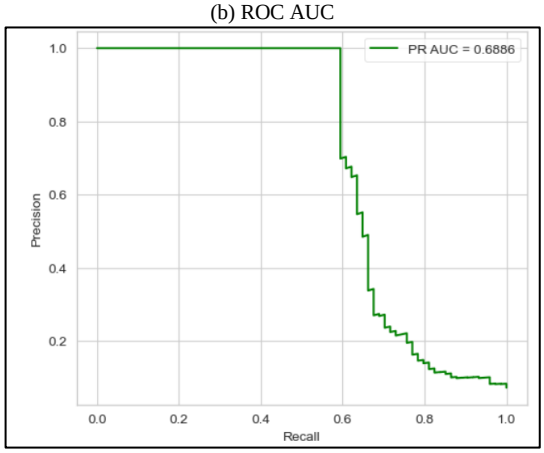
The confusion matrices of the four models reveal a consistent reduction in the number of false positives (cases where "No Stroke" was incorrectly classified as "Stroke") (Fig. 25). The SVM model had 9 false positives, whereas the Random Forest model reduced this to 4. The hybrid model (SVM + RF) further reduced the number to 2. The tuned hybrid model achieved zero false positives, correctly classifying all 942 "No Stroke" cases. Importantly, the number of correctly classified "Stroke" cases remained consistent at 44 across all models, and the number of False Negatives (Stroke cases misclassified as "No Stroke") remained at 30.



(a) Confusion Matrix



(b) ROC AUC



(c) PR AUC

Fig. 23. Performance evaluation after hyperparameter tuning.

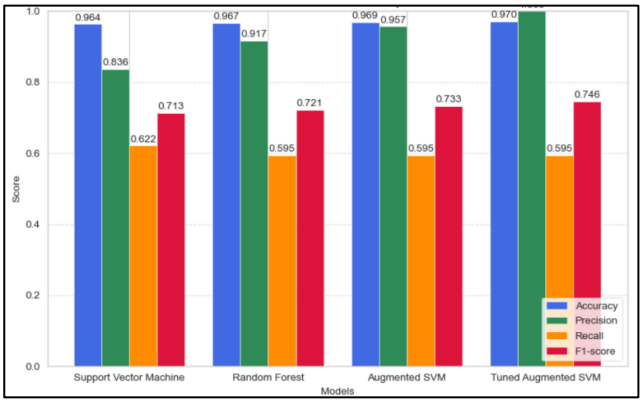


Fig. 24. Performance evaluation among four models.

According to Fig. 26(a), the tuned hybrid model achieved an AUC score of 0.842, slightly lower than the Random Forest (0.889) and SVM (0.875) models. However, it demonstrated a significant advantage by eliminating all false positives, as indicated in the confusion matrix, correctly classifying all 942 "No Stroke" cases. Despite a minor decrease in AUC, the tuned hybrid model maintained consistent performance in identifying true stroke cases, achieving a balanced outcome between sensitivity and specificity. In the performance comparison of Fig. 26(b), the Random Forest model achieved the highest PR AUC of 0.724, followed by the SVM with a PR AUC of 0.709, and the hybrid model with a PR AUC of 0.703. The tuned hybrid model achieved a PR AUC of 0.688, showing a slight drop compared to the other models.

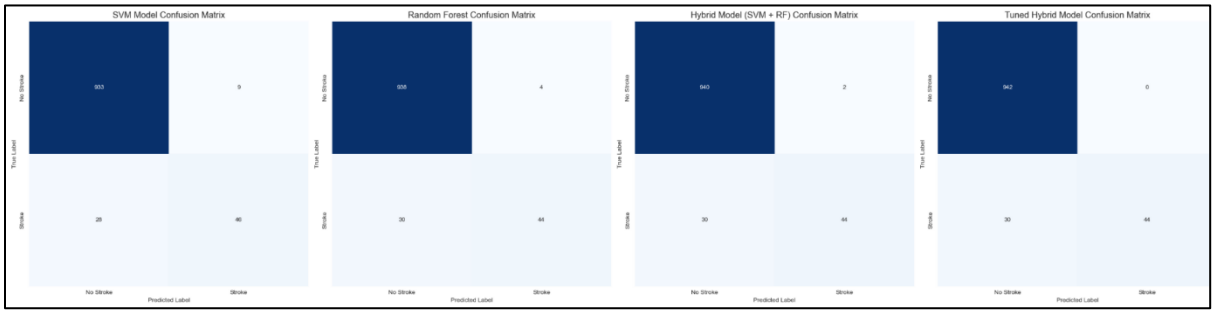
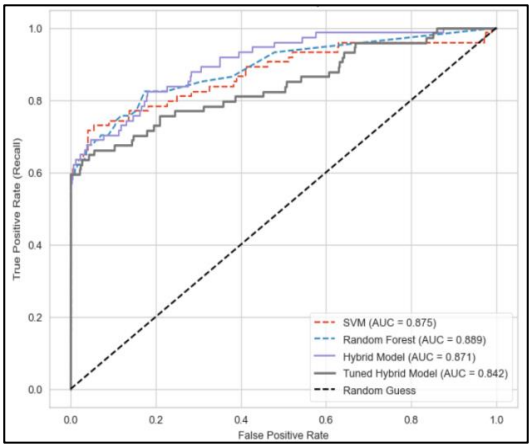
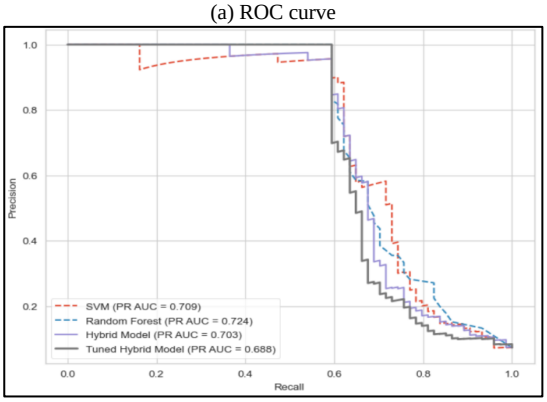


Fig. 25. Confusion matrices of four models.



(a) ROC curve



(b) Precision-Recall curve

Fig. 26. Performance evaluation (ROV curve and Precision Recall curve) for four models.

V. Discussion

In this project, the metrics used to evaluate the model performance included accuracy (overall correctness), precision (the proportion of predicted positives that are actually correct), recall (the proportion of actual positives correctly identified), F1-score (the harmonic mean of precision and recall), confusion matrix (detailed breakdown of predictions), ROC AUC (model's ability to distinguish between classes), and PR AUC (trade-off between precision and recall, particularly valuable for imbalanced datasets).

The confusion matrix highlights four key components: True Positive (TP), where the model correctly predicts a positive case; False Negative (FN), where the model correctly predicts a negative case for a positive instance; True Negative (TN), where the model correctly predicts a negative case; and False Positive (FP), where the model incorrectly predicts a positive case for a negative instance.

The standalone SVM achieved an accuracy of 96.36%, a precision of 83.64%, a recall of 62.16%, an F1 score of 71.32%, an ROC AUC of 0.8418, and a PR AUC of 0.6688, with a confusion matrix of 933 TN, 9 FP, 28 FN, and 46 TP. For Random Forest (RF), the performance improved with an accuracy of 96.75%, a precision of 93.62%, a recall of 59.46%, an F1 score of 72.73%, an ROC AUC of 0.8891, and a PR AUC of 0.7192, with a confusion matrix of 938 TN, 4 FP, 30 FN, and 44 TP. The feature-augmented model achieved 96.85% accuracy, 95.65% precision, 59.46% recall, 73.33% F1 score, 0.8707 ROC AUC, and 0.7038 PR AUC, with 940 TN, 2 FP, 30 FN and 44 TP.

The feature-augmented model demonstrated an improvement in accuracy, precision and F1-score, particularly in identifying stroke cases. Despite a slight decrease in the ROC AUC and PR AUC scores, the reduction in false positives (from 9 and 4 to 2) suggests that the model is more reliable, with fewer false alarms. This makes the model more robust for real-world deployment, especially where false positives could lead to unnecessary tests, anxiety, and increased healthcare costs.

To further enhance model performance, a Random Search hyperparameter tuning was applied. After tuning, the model achieved 97.05% accuracy, 100% precision, 59.46% recall, 74.58% F1-score, 0.8421 ROC AUC, and 0.6886 PR AUC. The confusion matrix showed 942 true negatives, 0 false positives, 30 false negatives, and 44 true positives.

The tuned feature-augmented model showed improvements in accuracy, precision and F1 score, although recall remained the same. The reduction of false positives to 0 and the slight increase in true negatives (from 940 to 942) indicate that the tuning process further optimised the model. This highlights the importance of minimising false alarms, particularly in medical applications, as false positives can result in unnecessary tests, increased anxiety, and additional healthcare costs.

This final tuned model provides a reliable and precise tool for stroke prediction, making it highly applicable in clinical environments where timely and accurate decision-making is essential. By reducing false positives and improving prediction reliability, the system is well suited for deployment in real-world applications, ensuring better health outcomes.

VI. Conclusions

In this project, a stroke prediction system is developed and evaluated using machine learning techniques. The hybrid model, combining Support Vector Machines (SVM) and Random Forest (RF), demonstrated a significant improvement over standalone models, achieving better accuracy, precision, and F1 score. The feature-augmented model, which incorporated additional features, further enhanced predictive performance, showing improved classification metrics, including a reduction in false positives and an increase in true negatives.

Through the process of hyperparameter tuning using Random Search, the model's performance was further optimised. The final tuned model achieved a high level of precision (100%), improved accuracy (97.05%), and maintained the same recall (59.46%) compared to the original feature-augmented model. Although there was a slight reduction in the ROC AUC and PR AUC, the ability to minimise false positives and improve true negatives makes it more reliable and suitable for real-world deployment in medical applications.

The practical implications of this project are significant, as it shows the potential of machine learning models to improve stroke prediction and aid healthcare providers in identifying high-risk patients. By reducing false alarms and increasing prediction accuracy, the model can help streamline healthcare processes, reduce unnecessary tests, and ultimately lower healthcare costs. This system has the potential to revolutionise stroke prediction by enabling early detection and intervention, thus significantly reducing the burden of stroke-related healthcare challenges globally.

Despite the strong performance, there are a few limitations. A major challenge is class imbalance, which is common in medical data sets, where positive stroke cases are significantly fewer than negative cases. This imbalance can affect recall, as seen with the relatively lower recall rate of 59.46%, suggesting that some true stroke cases are still being missed. There is also room for improvement in recall, which would ensure fewer missed stroke cases. Although high precision is critical, improving recall is equally important to ensure that at-risk patients are not overlooked.

To enhance the performance and utility of the stroke prediction system, several directions for future development can be considered.

- Incorporating additional features: Including more comprehensive patient data, such as laboratory test results, imaging data, or genetic markers, could improve the model's ability to detect subtle risk factors.
- Handling imbalanced datasets: Given the natural imbalance in stroke datasets (fewer stroke cases than non-stroke cases), applying techniques such as SMOTE (Synthetic Minority Oversampling Technique), ADASYN, Random Oversampling, or cost-sensitive learning can help improve recall without sacrificing precision.
- Exploring advanced ensemble techniques: Approaches like Gradient Boosting Machines (e.g. XGBoost, LightGBM) or stacking multiple models could potentially yield better performance by capturing complex patterns more effectively.

Funding

The APC of this paper publication is sponsored by INTI International University and Asia Pacific University.

Conflict of Interest

The authors declare that there is no conflict of interest.

References

[1] K. S. Tan and N. Venketasubramanian, "Stroke Burden in Malaysia," Cerebrovasc Dis Extra, vol. 12, no. 2, pp. 58–62, Mar. 2022, doi: 10.1159/000524271.

[2] S. Afzal et al., "Evaluating eating patterns and health status of undergraduate students majoring in human nutrition, Lahore, Pakistan: a cross-sectional study," Int J Adolesc Youth, vol. 30, no. 1, Dec. 2025, doi: 10.1080/02673843.2024.2448287.

[3] C. S. Nwosu, S. Dev, P. Bhardwaj, B. Veeravalli, and D. John, "Predicting Stroke from Electronic Health Records," in 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, Jul. 2019, pp. 5704–5707. doi: 10.1109/EMBC.2019.8857234.

[4] T. T. Tin, E. H. C. Sheng, L. S. Xian, L. P. Yee, and Y. S. Kit, "Machine learning classification of rainfall forecasts using Austin weather data," International Journal of Innovative Research and Scientific Studies, vol. 7, no. 2, pp. 727–741, Mar. 2024, doi: 10.53894/ijirss.v7i2.2881.

[5] J. El, "Historical Developments of Random Forest - Jari El - medium. Medium."

[6] T. T. Tin, L. S. Hock, and O. M. Ikumapayi, "Educational Big Data Mining: Comparison of Multiple Machine Learning Algorithms in Predictive Modelling of Student Academic Performance," International Journal of Advanced Computer Science and Applications, vol. 15, no. 6, 2024, doi: 10.14569/IJACSA.2024.0150664.

[7] W. Jeff, "Support Vector Machines." Accessed: Sep. 29, 2025. [Online]. Available: https://www2.isye.gatech.edu/~jeffwu/isye8813/SVM.pptx

[8] M. Shabil et al., "Association between hydrocarbon exposure and risk of stroke: a systematic literature review," BMC Neurol, vol. 25, no. 1, p. 71, Feb. 2025, doi: 10.1186/s12883-025-04083-x.

[9] P. Bentley et al., "Prediction of stroke thrombolysis outcome using CT brain machine learning," Neuroimage Clin, vol. 4, pp. 635–640, 2014, doi: 10.1016/j.nicl.2014.02.003.

[10] H.-L. Wang et al., "Automatic Machine-Learning-Based Outcome Prediction in Patients With Primary Intracerebral Hemorrhage," Front Neurol, vol. 10, Aug. 2019, doi: 10.3389/fneur.2019.00910.

[11] A. A. Abujaber et al., "Predicting 90-day prognosis for patients with stroke: a machine learning approach," Front Neurol, vol. 14, Dec. 2023, doi: 10.3389/fneur.2023.1270767.

[12] S. Rahman, M. Hasan, and A. K. Sarkar, "Prediction of Brain Stroke using Machine Learning Algorithms and Deep Neural Network Techniques," European Journal of Electrical Engineering and Computer Science, vol. 7, no. 1, pp. 23–30, Jan. 2023, doi: 10.24018/ejece.2023.7.1.483.

[13] H. Zhang, "Stroke Prediction Based on Support Vector Machine," Highlights in Science, Engineering and Technology, vol. 31, pp. 53–59, Feb. 2023, doi: 10.54097/hset.v31i.4812.

[14] Y. Wu and Y. Fang, "Stroke Prediction with Machine Learning Methods among Older Chinese," Int J Environ Res Public Health, vol. 17, no. 6, p. 1828, Mar. 2020, doi: 10.3390/ijerph17061828.

[15] V. Bandi, D. Bhattacharyya, and D. Midhunchakkravarthy, "Prediction of Brain Stroke Severity Using Machine Learning," Revue d'Intelligence Artificielle, vol. 34, no. 6, pp. 753–761, Dec. 2020, doi: 10.18280/ria.340609.

[16] M. S. Azam, M. Habibullah, and H. K. Rana, "Performance analysis of various machine learning approaches in stroke prediction," Int J Comput Appl, vol. 175, no. 21, pp. 11–15, 2020.

[17] Md. M. Islam, S. Akter, Md. Rokunojjaman, J. H. Rony, A. Amin, and S. Kar, "Stroke Prediction Analysis using Machine Learning Classifiers and Feature Technique," International Journal of Electronics and Communications Systems, vol. 1, no. 2, pp. 57–62, Dec. 2021, doi: 10.24042/ijecs.v1i2.10393.

[18] M. Alruily, S. A. El-Ghany, A. M. Mostafa, M. Ezz, and A. A. A. El-Aziz, "A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction," Applied Sciences, vol. 13, no. 8, p. 5047, Apr. 2023, doi: 10.3390/app13085047.

[19] R. Ren, H. Luo, C. Su, Y. Yao, and W. Liao, "Machine learning in dental, oral and craniofacial imaging: a review of recent progress," PeerJ, vol. 9, p. e11451, May 2021, doi: 10.7717/peerj.11451.

[20] B. Khadka, "Enhancing cloud based software engineering with machine learning," Master of Engineering, Information Technology, Centria University of Applied Science, 2024.

[21] L. Yang et al., "MRI classification using semantic random forest with the auto-context model." Accessed: Sep. 29, 2025. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8611460/pdf/qims-11-12-4753.pdf

[22] M. Nirmala and V. Saravanan, "Clinical Implication of Machine Learning Based Cardiovascular Disease Prediction Using IBM Auto AI Service," Int J Res Appl Sci Eng Technol, vol. 10, no. 8, pp. 124–144, Aug. 2022, doi: 10.22214/ijraset.2022.46087.

[23] L. K. Ramasamy, S. Kadry, Y. Nam, and M. N. Meqdad, "Performance analysis of sentiments in Twitter dataset using SVM models," International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 3, p. 2275, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2275-2284.

[24] N. Saxena et al., "Hybrid KNN-SVM machine learning approach for solar power forecasting," Environmental Challenges, vol. 14, p. 100838, Jan. 2024, doi: 10.1016/j.envc.2024.100838.

[25] D. Mustafa Abdullah and A. Mohsin Abdulazeez, "Machine Learning Applications based on SVM Classification A Review," Qubahan Academic Journal, vol. 1, no. 2, pp. 81–90, Apr. 2021, doi: 10.48161/qaj.v1n2a50.

Socio-Technical Factors Influencing Business Intelligence Adoption in SMEs

Ibrahim Abdusalam Abubaker Alsibhawi, Hazura Binti Mohamed, Jamaiah Binti Yahaya
Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Malaysia

***Abstract*—This study explores the major challenges that Small and Medium-sized Enterprises (SMEs) encounter when adopting Business Intelligence Systems (BIS), particularly in complex socio-political environments, such as Libya. It aims to understand how internal constraints, like limited financial capacity, resistance to change among management, and weak knowledge-sharing practices, combined with external socio-political factors, influence BIS adoption in developing economies. A cross-sectional survey approach was employed, targeting 297 SME owners and managers in Libya. Data was collected using a structured questionnaire and analyzed with SmartPLS to examine the relationships among key variables: facilitating conditions, information quality, perceived ease of adoption, perceived usefulness, and social influence. The findings highlight that social influence, especially from peers and industry experts, plays a crucial role in shaping SMEs' adoption behavior. Moreover, the quality of information emerged as a significant determinant in the successful adoption of BIS. The study offers both practical and policy-level insights, suggesting that with the right support, BIS adoption can significantly enhance SMEs' competitiveness, decision-making capabilities, and operational efficiency.**

Keywords*—*Information quality; social influences; perceived usefulness of Business Intelligence Adoption; perceived ease of adoption of Business Intelligence System; Business Intelligence System adoption

I. Introduction

In today's fast-paced and ever-changing economic environment, organizations are compelled to utilize tools that support optimal decision-making and help them adapt effectively without incurring financial losses. One growing trend is the adoption of Business Intelligence Systems (BIS), which are recognized for giving companies a significant competitive edge. A study by Tantawy and Ismail (2021) examined a large retail company operating in multiple countries that implemented BIS to enhance decision-making, operational efficiency, and customer satisfaction. The implementation led to notable performance improvements, including increased revenue, better inventory management, and improved customer segmentation. In 2017, Holiday Retirement collaborated with Prorize LLC to develop a revenue management system utilizing operations research. This system provided optimal pricing recommendations across more than 300 senior living communities, resulting in consistent revenue increases of over 10%. In this same vein, Continental Airlines reported increased revenues and cost savings, resulting in a 100% return on investment (ROI). The key to such success lies in accessing accurate and timely information to guide strategic decisions [59]. However, as noted by [10], poor BI adoption can lead to adverse operational impacts, including loss of market share and a declining customer base.

Nevertheless, some studies emphasized that knowledge serves as a valuable organizational resource capable of generating wealth. However, simply possessing knowledge is insufficient; it must be directed and applied effectively to support sound decision-making. Organizations can enhance their existing knowledge base through the implementation of Business Intelligence (BI) initiatives. The success of such initiatives is influenced by various elements, including the quality of information available [16], the ease with which systems can be used [85], and their perceived benefits. Numerous studies have highlighted that the most crucial contributors to the successful implementation of BI systems are related to how organizations manage their information technology, particularly through practices like knowledge sharing and managing organizational change.

In the case of Libyan SMEs, there exists a lack of sufficient knowledge about BIS and its potential benefits, which hinders informed decision-making regarding their adoption. Many Libyan SMEs face resistance to adopting new technologies due to a preference for traditional business practices and a lack of change management strategies. These and many challenges require targeted strategies that will provide clear key success factors for BIS adoption.

Despite the global recognition of BIS as a driver of competitiveness, the adoption rate among SMEs in developing economies—particularly in Libya—remains considerably low. Limited infrastructure, inadequate technical expertise, and cultural resistance to digital transformation exacerbate the challenge. This creates a gap between the proven benefits of BIS in advanced markets and the unrealized potential within Libyan SMEs. Consequently, the absence of evidence-based strategies tailored to this context leaves decision-makers without practical guidance on how to overcome implementation barriers.

This research, therefore, addresses a critical problem: identifying the underlying factors that influence BIS adoption in Libyan SMEs. By investigating organizational, technological, and environmental conditions that either enable or constrain adoption, the study seeks to bridge the gap between theoretical promise and practical application. The objective is to provide a framework that highlights key success factors and offers actionable insights for SMEs, policymakers, and industry stakeholders aiming to enhance competitiveness through digital transformation.

The significance of this study lies not only in its contribution to academic discourse but also in its practical relevance. For Libyan SMEs, understanding how to leverage BIS effectively could improve efficiency, foster innovation, and support long-term sustainability in an increasingly competitive marketplace. For policymakers, the findings will offer evidence-based recommendations to design supportive policies and capacity-building programs that encourage technological adoption across the SME sector.

To achieve these aims, the paper is structured as follows: following this introduction, a literature review examines existing studies on BIS adoption and highlights the gaps within the Libyan context. The methodology section then outlines the research design, data collection, and analysis techniques employed. This is followed by the presentation and discussion of findings, where key success factors are identified and critically examined. The paper concludes by summarizing the implications for practice and policy while suggesting directions for future research.

Based on this, the current research aims to identify the key success factors for BIS adoption, emphasizing their significance in the context of Libyan Small and Medium Enterprises (SMEs).

II. Literature Review

A. Business Intelligence System

Business Intelligence System (BIS) has been seen as a product that is made up of several processes, such as perception, storage, learning, communication, and decision [14]. Accordingly, there are various views on BIS in terms of gathering, processing, interpreting, and communicating the necessary information in the decision-making processes, and with a similar approach. One could conclude that this conceptualizes recognizing the value of external environmental information for shaping organizational actions through managing the information life cycle, which includes perspectives from information systems and their impact context, enhancing value-added informational processes, and facilitating decision-making processes.

Within the context of small and medium-sized enterprises (SMEs), intelligence is not simply a characteristic or inherent quality, but rather a dynamic capability that develops at individual, group, and organizational levels. This development is shaped by interactive communication and relationships across these levels, particularly through problem-solving and actions that support the creation and transfer of knowledge. [14] define business intelligence as an organization's ability to respond to new challenges and tasks, and to strategically adapt by generating and utilizing knowledge. In this view, intelligence is a capacity that enables organizations to adjust to external changes, while the creation and application of knowledge underpin both business intelligence and strategic decision-making. However, success and long-term viability of SMEs in competitive markets rely heavily on the ability to extract valuable insights from data, recognize patterns, and identify trends to make informed decisions [15]. In the past, SMEs lacked the technological infrastructure to process information efficiently, resulting in strategic, tactical, and operational decisions being made largely on intuition. Over time, certain automated systems capable of receiving raw data and quickly distributing relevant information to appropriate users was conceived. This system utilizes methods such as automatic document summarization, encoding, and user profile updates. But, these early techniques, primarily statistical in nature, were limited in effectiveness without adequate communication systems and input/output tools. Gradually, the proposed approaches evolved, giving rise to advanced concepts and methodologies designed to enhance decision-making based on factual data and led to the broader development of what became known as "Business Intelligence System" (BIS), a term later coined by [29]. Today, BIS encompasses a wide range of applications, technologies, and methods that facilitate the collection, organization, transformation, and analysis of structured data. This process can involve direct access tools like queries and reports, or more complex analytical techniques that turn raw data into actionable knowledge [85].

However, the effectiveness of BIS relies not only on advanced methodologies and technologies but also on understanding both internal and external factors, such as economic, environmental, market behavior, legislation, currency exchange rate, internal communication, and operations that affect an organization (Singh, Thakur, & Sharma, 2016). These factors generally influence all stakeholders in a business (customers, competitors, partners, and many more), collectively guiding business decisions and enabling the creation of significant competitive advantages. According to [23], "The best way to analyze the environment is not to try to understand it as a collection systems and organizations, but to treat it as information that the company must access through research activities." He also emphasized that "what truly matters is not just the suppliers or customers themselves, but the information that the company possesses about their goals, interests, relationship conditions with SMEs, and many other aspects of their behavior. This information helps identify their characteristics and expectations for the SMEs. Pas studies similarly asserted "The rapid pace of product innovation and market changes requires that future firms have a clear and expensive view of the business environment." In response to tight deadlines, successful SMEs must proactively explore new applications for emerging technologies, initiate product development ahead of demand, cultivate and stimulate demand, and establish market penetration strategies in anticipation of future needs. Additionally, SMEs must develop a high sense of anticipation of possible agreements or technological developments of rival companies. The use of business intelligence has evolved by leaps and bounds in proportion to the rise of information technology and the growth of the global industry. Experts in the field have provided numerous definitions of business intelligence. It is often described as the process of transforming data into knowledge to obtain a competitive advantage.

Past studies identified the direct impact of business intelligence on startup business performance, as well as its indirect impact through network learning and innovativeness [16]. Their results revealed a significant and positive impact of business intelligence on innovativeness, network learning, and startup business performance. Hence, focusing on the role of BIS can enhance organizational network learning and

performance. On the other hand, [14] found that BIS significantly enhances organizational performance by facilitating business process changes. Similarly, [83] found a significant and positive impact of business intelligence on organizational performance, with this impact fully mediated by the alignment of business intelligence strategies.

Nevertheless, it is necessary to indicate that Libyan SMEs face certain challenges related to data and information which represent BIS and have the potential to affect future success in making use of adequate processes of strategic planning, planning by scenarios or mixed approaches, or to achieve flexible and scalable management against the continuous growth changes of companies, without losing the characteristics of efficiency, quality, and the ability to make decisions. For example, many Libyan SMEs struggle with limited access to reliable market data and lack the technological infrastructure needed for efficient data management and strategic forecasting. Importantly, it is necessary to differentiate between the terms data, information and knowledge, as they are sometimes used interchangeably in literature. The data is a symbolic representation, which has no semantic meaning; it does not transmit a message. On the other hand, information is the set of processed and organized data that has a meaning, transmits a message, allows decision-making, favors problem-solving, and increases knowledge. For the information, the source is the data, and the required activity is the structuring and interpreting of the data. With respect to knowledge, the origin is given in the information that requires analysis, synthesis, dialectical vision, and determination of inferences. Also, the high competition among Libyan SMEs, driven by the need to increase sales rates and market share, necessitates maintaining effective internal management inform of BIS as explained in the current study. In this context, BIS includes innovation, creativity, continuous growth, competitiveness, and decision-making, all within a systematic spiral of continuous improvement. This and many more signify the urgent solution required by the Libyan SMEs towards having a dynamic and globalized environment, information, knowledge, methodologies, and management strategies generate greater demands on executives and managers, impacting the overall performance of the organization [85]. As such, the current study investigates the key factors that influence the adoption of Business Intelligence Systems (BISs) within Small and Medium-sized Enterprises (SMEs).

B. Technology Acceptance Model (TAM)

The use and acceptance of information technologies by end users is an additional way for the system to be successful [4]. Many theories have addressed the use of information technologies by individual consumers, such as the Technology Acceptance Model (TAM), which represents the most widely used [56, 66, 11]. The Technology Acceptance Model (TAM), originally developed by [24], is grounded in the Theory of Planned Behaviour by [31. Its primary objective is to elucidate and predict individuals' acceptance and utilization of technological innovations. The initial model was refined and expanded upon by [24], solidifying its place in academic literature.

TAM posits that those two critical factors, perceived usefulness (PU) and perceived ease of use (PEOU), significantly influence users' attitudes toward technology adoption, as shown in Fig. 1. These perceptions shape behavioural intentions, ultimately affecting actual technology usage. Recent studies have extended TAM by integrating additional variables such as perceived enjoyment, self-efficacy, and perceived cyber risk, enhancing its predictive power in diverse contexts. Initially, TAM was predominantly applied to organizational information systems, focusing on professional users. However, recognizing that general users may lack the technical expertise of professionals, researchers have advocated for model adaptations that incorporate humanistic and social factors, as in the current study. For instance, the inclusion of constructs like personal innovativeness and perceived enjoyment has been shown to better capture end-user behaviours [30].

Contemporarily, adopting and using information technologies in the business environment has been the main focus of information systems research and applications. Understanding the conditions underlying which information systems will be owned by organizations and creating these conditions becomes a high priority, making the current research eminent to identify other social aspects that can contribute to technology to serve as BIS [74].

This concept has a rooted relevance in the theory of planned behavior, which adds that an emotion or personal predisposition of the user gives advance notice of future behaviors, adopting an intermediate role between the user's perceptions and the user's final decision. The current situation of the Libyan SMEs shows that many are facing challenges in implementing or adapting new functionalities of new technologies, and providing strategies for successful adoption by users. Based on this, the present study uses TAM to identify the intention of use and assess the acceptance of technology in different contexts.

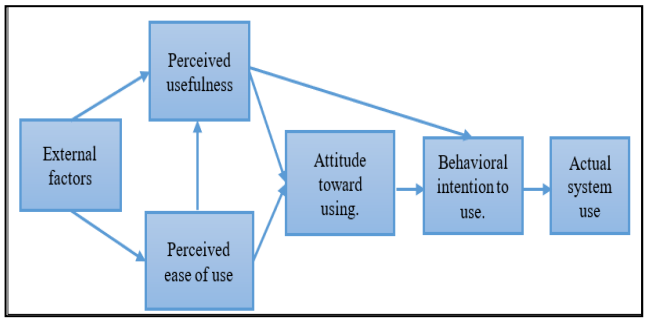


Fig. 1. Technological acceptance model. Source: [25].

C. Unified Theory of the Adoption and Use of Technology (UTAUT)

The Unified Theory of Acceptance and Use of Technology (UTAUT) model, developed by [10], integrates elements from various technology acceptance theories to explain user intentions and behaviors regarding technology adoption. Key constructs include performance expectancy, effort expectancy, social influence, and facilitating conditions, which directly influence behavioral intention and usage behavior were incorporated as shown in Fig. 2 below. Moderating factors such

as age, gender, experience, and voluntariness of use further refine the model's predictive capabilities.

Specifically, UTAUT was employed to address the Technology Acceptance Model (TAM) limitations by incorporating social and environmental factors. Recent studies have applied UTAUT across diverse contexts, including education, healthcare, and mobile technology, especially during the COVID-19 pandemic. For instance, [21] conducted a systematic literature review, revealing that performance expectancy was the most significant predictor of technology adoption during this period. Additionally, facilitating conditions, effort expectancy, and social influence were confirmed as influential factors in user adoption behaviors.

Further research by [25] applied UTAUT to personalized learning systems, finding that perceived usefulness, perceived ease of use, habit, and facilitating conditions significantly impacted both behavioral intention and actual system use. These findings underscore the model's versatility and relevance in understanding technology adoption in various settings. Based on our study, we posit that UTAUT would enhance the adoption of BIS among Libyan SMEs.

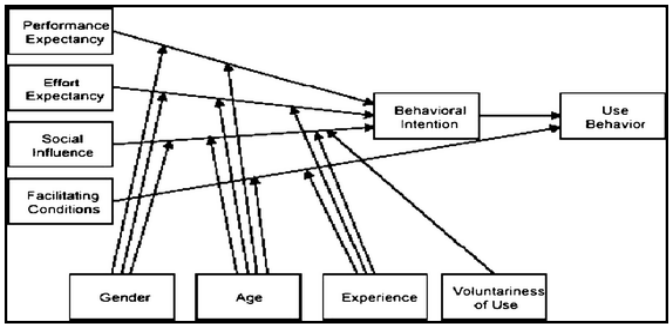


Fig. 2. Unified theory of adoption and use of technology, Source: [19].

D. The SMES Industry in Libya

SMEs in Libya are very important for the economic growth and social development of the country, in the contribution to the Gross Domestic Product (GDP) and job creation. Due to their characteristics, capacities and internal dynamics, they have attracted the attention of national and international organizations, authors, and researchers aiming to describe and address their challenges. The significant contribution of the SMEs industry to the Libyan economy motivated this study to investigate factors influencing the adoption of BIS. This technology is expected to help the SMEs industry achieve better performance and expand its market share within Africa.

SMEs are defined as businesses with limited employees and revenue, specifically those with fewer than 50 employees and less than USD700,000 in revenue, according to the Libyan regulation [5]. In [77], note that SMEs have distinct characteristics and dimensions but face occupational and financial limitations imposed by the regulations of each country. Consequently, there is no universal definition or characteristic that fully captures what SMEs are. The term SME refers to small and medium-sized companies. Typically, SMEs lack a defined structure and are often managed solely by the owner, who handles decisions related to pricing, hiring, salaries. Additionally, SMEs do not have specific job descriptions, and employees often perform multiple roles.

Small and medium businesses have different definitions in different countries. Generally, a small business is an organisation with limited number of employees, and typically, these employees are below fifty. In Libya, Small Business Administration defines small businesses as firms employing 50 or fewer persons, while medium enterprises are firms employing 50 to 250 as shown in Fig. 3.

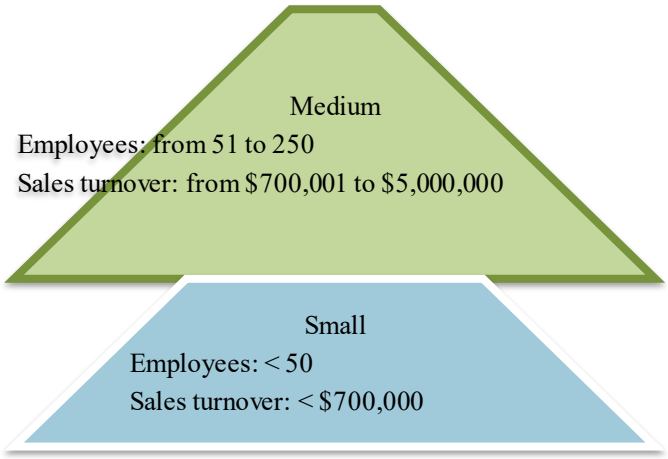


Fig. 3. SMEs classification in Libya. Source: Ministry of Industry - Libya 2020.

Currently, many Libyan small and medium businesses struggle due to the inadequate knowledge of business owners regarding the formulation of growth strategies. One of the reasons these businesses often fail to thrive is the low skill levels of their owners and managers [53]. According to [60], the absence of formal operational policies from authorities, which would guarantee the rights of informal business owners and managers, often leads to failures. In [55], the author pointed out that formalising business operations can lead to significant performance improvements for this business. In light, our study investigates the key factors that would influence the adoption of Business Intelligence Systems (BISs) within Small and Medium-sized Enterprises (SMEs) of Libya [77]. This is because it is very important for these SMEs to perform well, as the higher the number of successful SMEs can contribute to better Libyan economic conditions.

E. Research Hypotheses

The Technology Acceptance Model (TAM) posits that perceived usefulness is a key factor that influences the adoption and utilization of technology [25]-[28]. Information quality is a critical antecedent that can significantly shape this perception [44]. The better the quality of the information provided by BIS, the more likely the users are to find the system useful for decision-making and strategic planning [42]. Information quality can be characterized by attributes such as accuracy, timeliness, completeness, and reliability [12] [13]. Each of these facets contributes to the overall perception of usefulness. Previous studies underlined the importance of information quality for organizational performance [48], suggesting that good information quality enhances decision-making and increases perceived usefulness.

In sectors like healthcare, where the quality of information can be a matter of life and death, the link between information quality and the perceived usefulness of BIS is even more significant [48]. Similarly, in the context of SMEs struggling through unprecedented challenges such as the COVID-19 pandemic, the quality of information is paramount for managing crises and making informed decisions [41, 42]. Advancements in Artificial Intelligence (AI) have contributed to the improvement of information quality, subsequently affecting the perceived usefulness of systems like BIS. Studies like those by [57, 58] reflected that AI algorithms can enhance the quality of data mining, predictive analytics, and real-time data processing, thus making BIS more useful for end-users.

The relationship between information quality and perceived usefulness has been empirically tested in various domains. For instance, [44] specifically looked at Business Intelligence and Analytics (BIA) in micro, small- and medium-sized enterprises (MSMEs). They found that information quality directly correlates with the usefulness perceived by the businesses [45].

The hypothesis stands on strong theoretical and empirical foundations, making it a significant aspect of contemporary research in BISs. Quality of information is not just an isolated factor but rather a composite construct that profoundly influences how BIS is received and utilized in various organizational contexts. Based on the previous studies, such as [46, 78], which found a significant impact of information quality on the perceived usefulness of BIS, the proposed hypothesis is as follows:

H 1: There is a significant impact of information quality on the perceived usefulness of BIS

The concept of ease of implementation serves as a significant predictor for the acceptance and successful deployment of technological solutions, according to the TAM [25]. Information quality, with attributes such as accuracy, reliability, and timeliness, can significantly affect this ease [82]. The intersection of these two domains in the realm of BIS remains an area ripe for investigation.

Artificial Intelligence (AI) is increasingly important in automating data analysis and validation processes, enhancing information quality. Studies by [17, 34] demonstrated that AI can substantially ease the process of BIS implementation through predictive algorithms and real-time analytics, thereby providing high-quality information that is easier to integrate into existing systems. During times of crisis, such as the COVID-19 pandemic, the quality of information becomes crucial for quick decision-making and agile responses. [18] observed that the quality of information provided by AI systems significantly impacted the performance and ease of implementation of such technologies in the hospitality industry.

Previous studies have explored the importance of Industry 4.0 in enhancing operational efficiency [23, 30, 33]. Here, information quality is paramount as it directly influences how smoothly these digital transformations take place. The author in [40] addressed the role of AI in the retail industry, where the quality of information is vital for inventory management, customer relations, and logistics. When information quality is high, retail businesses find it much easier to implement and benefit from BIS and other AI-driven technologies. The author in [20], delved into scenario planning and business negotiations, suggesting that good information quality can help in predicting different scenarios more accurately, making the implementation of BIS easier for strategic planning and decision-making. Based on the previous studies, such as [40, 49] that found a significant impact of information quality on the perceived ease of implementation of BIS, the proposed hypothesis is as follows:

H 2: There is a significant impact of information quality on the perceived ease of BIS

The Technology Acceptance Model (TAM) originally emphasized the importance of perceived usefulness and ease of use in technology adoption [25, 14]. In [29], the author extended the TAM, incorporating social influences as a pivotal determinant in the Unified Theory of Acceptance and Use of Technology (UTAUT). Social influences include subjective norms, social identity, and system exposure, which can significantly impact the perceived usefulness of technological systems like BIS.

The author in [2] explored AI adoption in human resource management and highlighted that one of the significant factors impacting its effectiveness is the organizational perception of its utility. Often, this perception is moulded by the social norms within the HR department and the larger organizational culture. In [6], the author discussed the enterprise-level analysis of data analytics-oriented business intelligence technology. That research emphasized that social factors within an organization can either facilitate or hinder the effectiveness of these systems, thus affecting their perceived usefulness. In [10], the author considered AI and big data from ontological and communicative perspectives, suggesting that the social discourse surrounding these technologies influences their perceived utility. Such discourse is often shaped by organizational leaders and opinion makers, whose views are socially influential.

The author in [12], on the adoption of AI in hiring processes, found that social factors like management support and peer encouragement were influential in determining how useful the AI systems were considered to be. In [30], the author discussed AI governance and pointed out that social considerations, such as ethical and societal norms, can strongly impact how the usefulness of AI technologies, including BIS, is perceived within an organization. A previous study discussed the role of complementary resources and external readiness in SME performance. Social influences in the external business environment can significantly shape how useful a firm perceives its BIS, especially regarding its functionality in e-commerce platforms. Based on previous studies such as [1, 67] that found a significant impact of social influences on the perceived usefulness of BIS, the proposed hypothesis is as follows:

H 3: There is a significant impact of social influences on the perceived usefulness of BIS

Traditional models like the Technology Acceptance Model (TAM) have been extended to include social factors, like the

Unified Theory of Acceptance and Use of Technology (UTAUT), which identifies social influences as a significant predictor of technology adoption ease. In this context, social influences encompass subjective norms, social identity, and system exposure. In [72] the author explored how artificial intelligence could manage agricultural supply chain risks, especially during the COVID-19 pandemic. They found that perceived ease of implementation was substantially influenced by social forces, such as farmers' communities, collaborative knowledge, and even governmental directives. In [76], the author focused on how AI can transform library services in educational institutions. They note that the success of such implementation often rests on social factors like faculty support, student enthusiasm, and administrative will, which influence how easy the implementation is perceived to be.

The authors in [83,3] explored the need to adopt big data analytics in B2B organizations. Their model underlines the significance of social influences, such as competitive pressures and partner expectations, in gauging how straightforward it is to implement new technologies like BIS. In [4], the study provided a framework to realize strategic value from learning analytics. They identified that the perception of ease of implementation is influenced by the institution's culture and beliefs, which are, in turn, driven by social influences like academic advocacy or resistance. Past literature discussed artificial intelligence acceptance in the context of digital entrepreneurship. They proposed that social influences like peer networks, industry thought leadership, and market trends significantly impact entrepreneurs' perception regarding the ease with which AI can be implemented.

Based on previous studies, such as [69] and [84], which found a significant impact of social influences on the perceived ease of implementation of BIS, the proposed hypothesis is as follows:

H 4: There is a significant impact of social influences on the perceived ease of BIS

In technological acceptance models such as the Unified Theory of Acceptance and Use of Technology (UTAUT), facilitating conditions are considered a critical determinant of technology acceptance. These conditions often correlate positively with the perceived usefulness of the technology in question, as they can directly impact the technology's efficiency and effectiveness. The author in [55] discussed the role of facilitating conditions, like training and technology support in adopting virtual reality technology for sustainable quality in agriculture. The study showed that these conditions significantly improved farmers' perception of the technology's usefulness, providing insights into how facilitating conditions can also influence perceived usefulness in the domain of BIS.

Past research studied the adoption of AI mobile banking applications [19], focusing on facilitating conditions, such as user interface design and customer support. They found that these conditions were essential contributors to the perceived usefulness of mobile banking applications, suggesting similar implications for BIS. The author in [61] delved into how blockchain announcements affect stock value from a technology management perspective. The study revealed that factors like a supportive regulatory environment and internal blockchain expertise (facilitating conditions) significantly enhanced the perceived usefulness of blockchain. This finding can be paralleled in the context of BIS [62, 63].

In [68], the author studied the benefits and limitations of AI adoption in the pandemic-stricken Indian industry. The facilitating conditions, like government support and availability of skilled labour, played a considerable role in influencing the perceived usefulness of AI technologies, supporting the hypothesis in the realm of BIS. [71] investigated the factors affecting blockchain adoption in apparel supply chains. Their findings highlighted that facilitating conditions like technology infrastructure and stakeholder support played a vital role in determining the perceived usefulness of blockchain technologies. Based on previous studies such as [11, 65], which found a significant impact of facilitating conditions on the perceived usefulness of BIS, the proposed hypothesis is as follows:

H 5: There is a significant impact of facilitating conditions on the perceived usefulness of BIS

Facilitating conditions refer to the degree to which an individual believes that an organizational and technical infrastructure exists to support the use of the system. If these conditions are favourable, they can significantly reduce the perceived complexity of implementing a new technology, such as BIS. The author in [32] examined the role of technology readiness and awareness in influencing the intention to use analytical AI in services. The study showed that facilitating conditions like technical readiness significantly influenced the perceived ease of implementing AI solutions. These findings can be extrapolated to understand how facilitating conditions could affect the ease of implementing BIS.

In [37], the author conducted a systematic review identifying the drivers and barriers of Industry 4.0 technology adoption among manufacturing SMEs. Their study highlighted that facilitating conditions, such as adequate training and technical support, can significantly ease the process of adopting advanced technologies, thereby supporting the hypothesis in the context of BIS. The author in [39] investigated the consumers' adoption of artificial intelligence and robotics in the hospitality and tourism sector. Their research noted that factors like consumer familiarity with technology (a facilitating condition) significantly impacted the perceived ease of interacting with such technologies. These findings support the notion that facilitating conditions similarly influence the perceived ease of implementing BIS.

Past literature discussed the use of conversational interfaces for accessing business relational data structures. They pointed out that user-friendly interfaces (a form of facilitating condition) make accessing and navigating complex data structures easier, potentially simplifying the implementation of BIS [51]. In [58], the author explored the perceptions of Indian managers on the impact of convergent technologies on work and resultant organizational performance. The study underlined the importance of facilitating conditions like managerial support and the availability of resources, which significantly influenced the ease of implementing these technologies. Based

on previous studies such as [18], which found a significant impact of facilitating conditions on the perceived ease of implementation of BIS, the proposed hypothesis is as follows:

H 6: There is a significant impact of facilitating conditions on the perceived ease of BIS

The perceived ease of implementation can affect the perceived usefulness by lowering the barriers to effective utilization. When a BIS is perceived as easy to implement, users are more likely to perceive it as beneficial to their tasks. Such a system seamlessly integrates with existing workflows and enhances its perceived usefulness.

In [2], research on AI adoption by human resource management highlighted that ease of use directly influences the perceived usefulness of AI systems in HR tasks. This research implies that when BIS are easier to implement, HR professionals may find them more useful for data analytics, strategic decision-making, and employee management. The work by [3, 52] presented a compelling argument for the role of organizational absorptive capacity in enhancing the efficiency of BISs. This study indirectly supported the hypothesis by suggesting that an organization's ability to assimilate and utilize new knowledge (a factor that can facilitate ease of implementation) significantly impacts the efficiency and, by extension, the perceived usefulness of BIS.

The study of [6] provided an enterprise-level analysis of business intelligence technology effectiveness. Their study posited that ease of integration and implementation are vital parameters for evaluating the effectiveness of data analytics-oriented BISs. Therefore, their findings provide empirical backing for the proposed hypothesis. Previous research [9] emphasized the relevance of self-service business intelligence tools in university management, noting that ease of use positively influences perceived usefulness. This is particularly applicable in an academic setting where administrative staff and faculty may lack advanced technical skills.

In the research of [10], which delved into the ontological and communicative perspectives of AI and big data. Their research suggested that easy-to-implement AI solutions, seamlessly integrate big data analytics are perceived as significantly more useful in solving complex business problems. Based on the previous studies, such as [81, 67] that found a significant impact of perceived ease of implementation of BIS on the perceived usefulness of BIS, the proposed hypothesis is as follows:

H 7: There is a significant impact of the perceived ease of BIS on the perceived usefulness of BIS

Perceived usefulness can be considered a key driver for the successful implementation of BIS. This is predicated on the assumption that perceived benefits like enhanced decision-making, improved efficiency, and increased competitive advantage will encourage stakeholders to support the system's implementation. The study by [2], examined AI adoption in human resource management and found that the perceived usefulness of AI systems significantly influences their implementation. This observation could be extrapolated to BIS implementation, suggesting that perceived usefulness would directly and positively impact the success of BIS roll-out.

In a study by [3], Serrano and Koulouri (2021), the authors explored the effect of organizational absorptive capacity on BIS efficiency. Although the focus is on efficiency, the underlying message is that organizations are more likely to successfully implement systems they find useful, further supporting the hypothesis in question. The author in [7] provided insights into the evaluation of data analytics-oriented business intelligence technology's effectiveness at the enterprise level. Their work underscores the importance of perceived usefulness as a key evaluative measure, indirectly implying that systems deemed useful are more likely to be implemented successfully.

According to [8] study, which delved into the utility of self-service business intelligence tools in university management. The emphasis on the perceived usefulness of these tools in facilitating administrative tasks suggests that these systems are more likely to be successfully implemented if they are deemed useful by the end-users. In [10] research, which explored the implications of AI and big data in modern businesses. Their research emphasized that the perceived usefulness of these technologies has a substantial impact on their successful integration into existing business processes. Based on previous studies, such as [81] found a significant impact of the perceived usefulness of BIS on the BIS implementation, the proposed hypothesis is as follows:

H 8: There is a significant impact of the perceived usefulness of BIS on the perceived ease of BIS

Information quality is often characterized by attributes like accuracy, timeliness, completeness, and relevance. High-quality information is expected to increase the perceived usefulness of a BIS, as it enables better decision-making. On the other hand, perceived usefulness refers to the degree to which a person believes using a BIS would enhance their job performance. Therefore, the better the information quality, the higher the perceived usefulness, which positively impacts the implementation of the BIS.

The study of [6] delved into the effectiveness of BIS at the enterprise level. Although the paper did not directly address the mediating role of perceived usefulness, it emphasized the importance of system effectiveness, which is closely linked to information quality. High-quality information would inevitably make a BIS more effective and enhance its perceived usefulness, thus influencing its successful implementation. The author in [8] focused on the applicability of self-service BIS tools in university management. In a self-service setting, the quality of information becomes even more critical as users need to rely on the data provided for decision-making. This highlights the importance of information quality in influencing perceived usefulness and successful implementation.

The work by [10] explored the adoption of AI and big data in different sectors. These advanced technologies enhance information quality, making systems more useful and more likely to be successfully implemented. Thus, the perceived usefulness here serves as a mediator between the high-quality information provided by AI and big data and the actual implementation of BIS. Based on previous studies such as [24,

48, 81], found significant mediation of the perceived usefulness of BIS between information quality and the BIS implementation, the proposed hypothesis is as follows:

H9: There is significant mediation of the perceived usefulness of BIS between information quality and the BIS implementation

Information quality in a BIS encompasses attributes like accuracy, timeliness, relevance, and completeness. When information quality is high, users are more likely to find the system easier to implement as they can more effectively use the data for decision-making. Ease of implementation refers to the perceived ease with which a system can be integrated into existing processes or a new setup can be operationalized. The easier the implementation, the more likely the system will be implemented successfully.

The research by [6] focused on evaluating data analytics-oriented BIS technology effectiveness at the enterprise level. While their work did not explicitly look at the ease of implementation, it emphasized the role of effective BIS systems. Effective systems characterized by high-quality information are likely to have reduced implementation barriers, thereby proving the mediating role of ease of implementation. [2] delved into the adoption of AI by Human Resource Management and its effectiveness. When AI algorithms produce high-quality information, HR professionals find the system easier to use and implement, thus providing an indirect link between information quality and successful implementation.

The study by [8] focused on the relevance of self-service BIS in university management. The importance of ease of implementation is implicitly emphasized here; high-quality information makes self-service tools more straightforward and more intuitive to implement. The author in [10] discussed the ontological and communicative perspectives in AI and big data scenarios in modern businesses. Here, the quality of information gained from these advanced technologies would naturally affect how easy it is to implement these systems, which affects their adoption rate. Based on previous studies, such as [81, 49] that found significant mediation of the perceived ease of implementation of BIS between information quality and the BIS implementation, the proposed hypothesis is as follows:

H10: There is a significant mediation of the perceived ease of BIS between information quality and the BIS implementation

F. Research Hypotheses

In this research, the existing problem has been identified as the lack of a BIS to improve SMEs' performance in Libya. Possible solutions to this problem have been formulated through hypotheses as explained in the previous section. This study identified several organizational and technological factors, such as information quality, social influences and facilitating conditions as the independent variable (IV), while perceived usefulness of BIS and perceived ease of BIS represent the mediation variable (MV) respectively toward BIS adoption which is the dependent variable (DV). The nomination of these factors was based on related theories such as the technology acceptance model (TAM), the unified theory of adoption and use of technology (UTAUT. This shows in the below research model Fig. 4 for the current study.

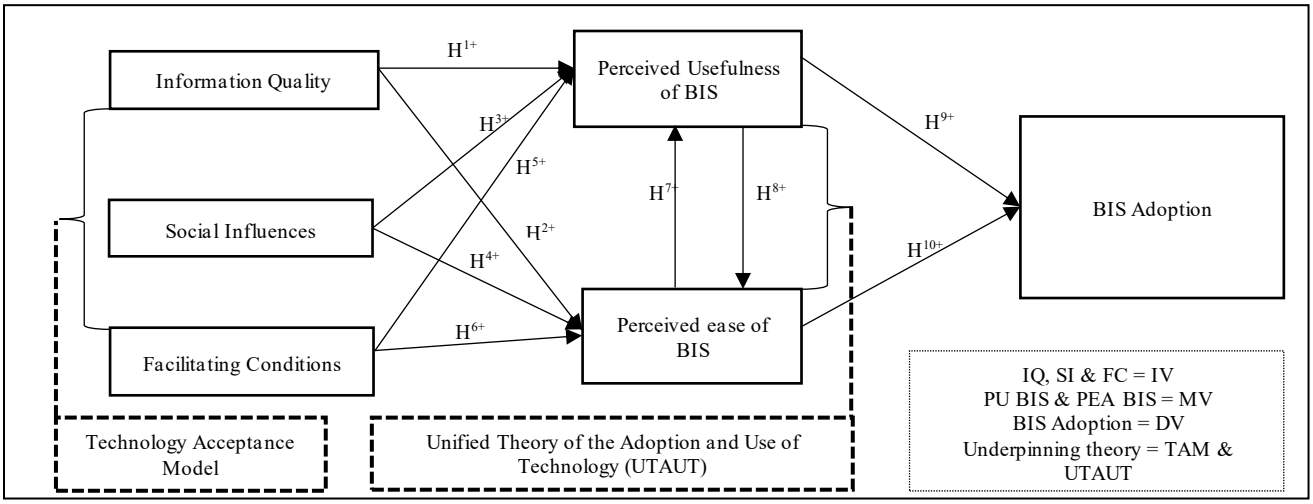


Fig. 4. Research model.

III. METHODOLOGY

The current study employed a quantitative methodology, which allowed to obtain relevant information to investigate factors that led Libyan SMEs towards adoption of BIS. According to [9], the research design clarifies the adopted process used to manage the data, which is used to figure out the possible solution to the research issues. The research design process includes several phases that lead to a potential solution for the research issues. These were identified by [22] in six phases as: 1) identifying the problem of the research, 2) underpinning the related theories and empirical studies, 3) clarifying the purpose of the research, 4) gathering the optimal data, 5) interpreting the obtained result retrieved from the data, and 6) conclude the finding of the research.

The population of this study represents the total management levels of those who work at Libyan SMEs all over the country. The sample size of this population is defined based on G-power software, which allows us to determine the optimal sample size based on the number of predictors within the constructed model proposed. The target population was 180,000 SMEs listed by the Libyan Ministry of Economy and Trade. Two methods were used to identify the target sample size based on the total population determined for this study: the optimal sample size proposed by [53]. Thus, the optimal sample size for this study was 384, as the total population is higher than 100,000. However, the response rate (77.3%) of the survey was 297 SMEs' owners and managers in Libya, which was seen as enough to test hypotheses based on numerical measurement and statistical analysis to derive a generalized result [35].

The questionnaire instrument was used for data collection as it provided measurements for the nominated factors. The design of the instrument comprises two main sections; the first section is about the respondents' profile, while the second section is assigned to measure the model construct. The study used SPSS27 and Smart PLS4 to test the relationships among the research variables:

A. Data Analysis and Results

In order to achieve the current research objective, the respondents' profile was analyzed with descriptive statistics. This is followed by the normality and the reliability test, respectively, as well as direct and indirect effects tests. The comprehensive details of each analysis are explained in the following sections.

B. Respondents Profile

It was important to determine the profile of respondents of the current study to better understand the target population, design research instruments, and draw more accurate conclusions from the data [75]. As such, several categories of respondents' profiles, such as gender, age, highest education, and working experience, were determined in the present study. Table I displays the profile of the respondents who participated in the study. The table shows that 71.4% of the participants belong to the male gender, while 28.6% belong to the female gender category. The result confirms that male respondents are the majority. In terms of the age of the participants, it was determined that most of the respondents (35.4%) were 20-31 years old, while 67 participants were in the range of 31-40 years old presenting 22.6% and 68 participants were in the range of 42-50 years old presenting 23.1%, with 57 participants found to be in age 51 years old and above 18.9%.

According to the results shown in Table I, most of the participants hold bachelor's certificates with a percentage of 32.8%. 28 participants hold high school certificates with a percentage of 9.4% and followed by 54 participants holding a diploma certificate with a percentage of 18.4%. Also, 60 participants hold master certificates with a percentage of 20.4% and 31 participants hold diploma certificates with a percentage of 10.5%, and 27 participants hold other certificates with a percentage of 8.5%. This ensures that all the participants are well educated [70]. The experience of the participants ranged from less than 1 year and more than five years. There are 15.2% of the participants who had experienced less than one year, followed by 23.7% of the participants with experience of 1 to 3 years, while 29.4% of the participants had experience of 3 to 5 years, and 31.7% of the participants had experience of more than five years. This shows that most of the respondents were well-experienced.

TABLE I. PROFILE OF RESPONDENTS (N = 297)

	n	%		n	%
Gender			**Highest Education**		
Male	212	71.4	High School	28	9.4
Female	85	28.6	Diploma	54	18.4
			Bachelor	97	32.8
Work Experiences			Master	60	20.4
Less than 1 year	45	15.2	PhD	31	10.5
1 to 3 years	70	23.7	Other	27	8.5
3 to 5 years	87	29.4	**Age**		
More than 5 years	95	31.7	21-30 years old	105	35.4
			31-40 years old	67	22.6
			41-50 years old	68	23.1
			51 years old and above	57	18.9

C. Descriptive Statistics

According to Table II, the mean statistics for all the items ranged between 2.811 and 4.387. Meanwhile, the standard deviation values ranged between 0.784 and 1.014. These results ensure that respondents are, on average agreement with the items stated for each factor in the questionnaire. Also, the results ensure the vital role of the independent and mediating variables in the BIS adoption by SMEs. The descriptive analysis for the current study also included the normality test. A normality test is a statistical procedure used to determine whether a given data set or sample is normally distributed or not. Normal distribution, also known as Gaussian distribution, is a bell-shaped probability distribution characterized by the mean, standard deviation, and other parameters that describe the spread and shape of the distribution.

The current research used the Skewness and Kurtosis values to test the normality of the distribution. The accepted range of Skewness between -3 and +3 is to be accepted, while the Kurtosis range should be between -5 and +5 [43]. Table II shows the result of the normality test for the construct that all the factors (information quality, social influences, facilitating conditions, perceived ease of implementation of BIS, perceived ease of implementation of BIS, and BIS adoption by SMEs) are normally distributed within the accepted range, with Skewness range between -1.646 and -0.014, as well as the Kurtosis range was between -0.563 and 3.818 [47].

D. Construct Reliability

The construct reliability test was carried out in this study to determine the internal consistency of the data [47]. Table III shows that the composite reliability, factor loadings, and Cronbach's alpha for each construct are more than the required threshold, which indicates that the measurements are reliable, where the factors (information quality, social influences, facilitating conditions, perceived ease of implementation of BIS, perceived ease of implementation of BIS, and BIS adoption by SMEs) were all within acceptable results. The Cronbach's alpha for the factors ranged between 0.754 and 0.898, while the composite reliability values ranged between 0.812 and 0.927 [78-80].

TABLE II. DESCRIPTIVE ANALYSIS

Items	Mean	Median	SD	Kurtosis	Skewness
IQ1	3.997	4.000	0.962	1.120	-1.087
IQ2	4.067	4.000	0.881	2.165	-1.200
IQ3	3.906	4.000	1.014	0.875	-1.017
IQ4	3.912	4.000	0.913	2.163	-1.212
SI1	3.781	4.000	0.874	0.198	-0.472
SI2	3.754	4.000	0.819	0.592	-0.480
SI3	3.953	4.000	0.844	0.671	-0.654
SI4	2.811	3.000	0.913	-0.563	-0.098
FC1	3.040	3.000	0.938	-0.158	-0.179
FC2	3.007	3.000	0.950	-0.385	-0.156
FC3	3.148	3.000	1.011	-0.448	-0.340
PU1	3.094	3.000	1.004	-0.376	-0.190
PU2	2.987	3.000	0.995	-0.243	-0.014
PU3	3.657	4.000	0.886	0.352	-0.877
PU4	3.838	4.000	0.730	2.707	-1.149
PEA1	3.596	4.000	0.909	1.631	-1.246
PEA2	3.485	4.000	0.821	1.559	-1.254
PEA3	4.387	5.000	0.784	3.818	-1.646
PEA4	4.232	4.000	0.827	2.789	-1.427
BIS1	4.131	4.000	0.978	1.589	-1.331
BIS2	3.808	4.000	0.833	0.585	-0.540
BIS3	3.700	4.000	0.784	0.952	-0.599

Where: IQ; Information quality, SI; Social influences, FC; Facilitating conditions, PU; Perceived usefulness of BIS, PEA; Perceived ease of adoption of BIS, and BIS; BIS adoption

E. Construct Reliability

The value used to test the convergent validity is the average variance extracted (AVE). According to the rule of thumb, when the value of AVE is greater than 0.5, the factor shows good construct validity. Table III shows that the factors (information quality, social influences, facilitating conditions, perceived ease of implementation of BIS, perceived ease of implementation of BIS, and BIS adoption by SMEs) have acceptable AVE values, ranging between 0.546 and 0.779.

The factor loading test was used in the study to confirm the validity of the research data. The accepted value for the factor loading is greater than 0.5. Table III reveals that the factors (information quality, social influences, facilitating conditions, perceived ease of implementation of BIS, perceived ease of implementation of BIS, and BIS adoption by SMEs) have acceptable factor loading values, ranging between 0.457 and 0.939.

F. Direct Effect Test

The direct effect test was employed in the study to examine the correlation between two variables by a correlation coefficient, whose value oscillates between -1 and +1. If the correlation coefficient is +1, it indicates a positive relationship between the variables, and -1 indicates a negative relationship between the two variables. Assessing path coefficients is an important step in conducting structural equation modeling (SEM), a statistical technique used to analyze complex relationships between variables. Path coefficients, also called standardized regression coefficients or beta weights, represent the strength and direction of the relationships between variables in a model [75]. For the current research, the path coefficients test was used to find out the relationships between the research variables.

TABLE III. ASSESSMENT OF THE FINAL MEASUREMENT MODEL

Model Construct	Items	Loading	CA	CR	AVE
BIS adoption	BIS1	0.823	0.754	0.857	0.667
	BIS2	0.855			
	BIS3	0.770			
Facilitating Conditions	FC1	0.927	0.864	0.913	0.779
	FC2	0.907			
	FC3	0.809			
Information Quality	IQ1	0.908	0.898	0.927	0.761
	IQ2	0.929			
	IQ3	0.782			
	IQ4	0.862			
Perceived Ease of Adoption of BIS	PEA1	0.457	0.757	0.812	0.546
	PEA2	0.467			
	PEA3	0.939			
	PEA4	0.936			
Perceived Usefulness of BIS	PU1	0.726	0.777	0.844	0.576
	PU2	0.703			
	PU3	0.791			
	PU4	0.809			
Social Influence	SI1	0.863	0.787	0.877	0.705
	SI2	0.901			
	SI3	0.748			

TABLE IV. DIRECT EFFECT

Paths	Beta	SD	T	P Values
Facilitating Conditions → Perceived Ease of Adoption of BIS	-0.010	0.051	0.189	0.850
Facilitating Conditions → Perceived Usefulness of BIS	0.758	0.029	25.881	0.000
Information Quality → Perceived Ease of Adoption of BIS	0.400	0.076	5.247	0.000
Information Quality → Perceived Usefulness of BIS	0.162	0.033	4.958	0.000
Perceived Ease of Adoption of BIS → BIS adoption	0.701	0.035	19.939	0.000
Perceived Usefulness of BIS → BIS adoption	0.031	0.052	0.596	0.552
Social Influence → Perceived Ease of Adoption of BIS	0.498	0.048	10.290	0.000
Social Influence → Perceived Usefulness of BIS	0.154	0.035	4.396	0.000

The Table IV above shows the direct effect of the hypotheses tested in the current study. The result of the Beta,

T-statistics and P-value (-0.010, 0.189, 0.850) show that there is no relationship between the facilitating conditions and the perceived ease of adoption of BIS as it is greater than the acceptable threshold of 0.05 [77]. However, other P-value of facilitating conditions and the perceived usefulness of BIS (beta = 0.758, T-value = 25.881, and P-value = 0.000), information quality and the perceived ease of adoption of BIS (beta = 0.400, T-value = 5.247, and P-value = 0.000), information quality and the perceived usefulness of BIS (beta = 0.162, T-value = 4.958, and P-value = 0.000), perceived ease of adoption of BIS and BIS adoption (beta = 0.701, T-value = 19.939, and P-value = 0.000), perceived usefulness of BIS and BIS adoption, social influence and the perceived ease of adoption of BIS (beta = 0.498, T-value = 10.290, and P-value = 0.000), social influence and the perceived usefulness of BIS with (beta = 0.154, T-value = 4.396, and P-value = 0.000) all shows that there is a positive and significant relationship as hypothesized.

G. Mediation Effect Test

The mediation test was used in the study to identify the mediation effect of perceived usefulness of BIS and perceived ease of adoption of BIS between the independent variables and the dependent variable as shown in Table V.

Table V shows that the perceived usefulness of BIS has no mediation relationship between facilitating conditions and BIS adoption as Beta=0.023, T-statistics=0.602 and P-value=0.547 all above or below the threshold values. Similarly, perceived ease of adoption of BIS has no mediating relationship between facilitating conditions and BIS adoption because Beta-value=-0.007, T-statistics=0.188 and P-value=0.851 all above or below the threshold values. However, perceived ease of adoption of BIS mediates the relationship between social influence and BIS adoption with (beta = 0.349, T-value = 8.636, and P-value = 0.000), while also perceived ease of adoption of BIS mediates the relationship between information quality and BIS adoption with (beta = 0.281, T-value = 4.984, and P-value = 0.000). Again, perceived usefulness did not mediate the relationship between information quality and BIS adoption as well as perceived usefulness of BIS did not mediate the relationship between social influence and BIS adoption as resulting in Beta = 0.005, T-statistics = 0.0576, P-value = 0.565 and Beta = 0.005, T-statistics = 0.0544, P-value = 0.587 all above or below the threshold values.

H. Discussion

The discussion section is the last step in the process of the findings. This section presents the results related to the research hypotheses and compares them with the results and findings of the previous studies. This study has found that BISs adoption factors in Libyan small and medium enterprises are facilitating conditions, information quality, perceived ease of adoption of BIS, perceived usefulness of BIS, and social influence. This study has found that there is no relationship between facilitating conditions and the perceived ease of adoption of BIS. There is a positive and significant relationship between facilitating conditions and the perceived usefulness of BIS with beta = 0.758, T-value = 25.881, and P-value = 0.000. There is a positive and significant relationship between information quality and perceived ease of adoption of BIS (beta = 0.400, T-value = 5.247, and P-value = 0.000). There is a positive and significant relationship between information quality and perceived usefulness of BIS (beta = 0.162, T-value = 4.958, and P-value = 0.000). There is a positive and significant relationship between the perceived ease of adoption of BIS and BIS adoption with beta = 0.701, T-value = 19.939, and P-value = 0.000. There is no relationship between the perceived usefulness of BIS and BIS adoption. There is a positive and significant relationship between social influence and perceived ease of adoption of BIS (beta = 0.498, T-value = 10.290, and P-value = 0.000). There is a positive and significant relationship between social influence and the perceived usefulness of BIS (beta = 0.154, T-value = 4.396, and P-value = 0.000).

These results were similar to the results of the previous studies, which means the previous studies' findings supported this study's findings. Previous research, such as [74,11] have indicated that both perceived usefulness (PU) and perceived ease of use (PEOU) directly impact the BIS implementation in general; these relationships will be needed in the context of BIS implementation, as stated within the original TAM model. Technology-driven strategy is defined by [36] as the development, integration, and usage of new technologies in new product and service development. Recent research has demonstrated that a technology-driven strategy strongly impacts developing information technology capabilities [43]. Therefore, the direct influence of the technology-driven strategy on BIS' perceived usefulness was tested. Information quality and organizational factors affect user satisfaction with technology and influence beliefs about using it [54]. The effect of information quality on PU and PEOU has been tested by [33, 50]. They found that information quality influences adoption success by positively impacting usefulness and perceived ease of implementation.

TABLE V. SUMMARY OF THE MEDIATION EFFECT TEST

Mediation Paths	Beta	SD	T	P-value
Facilitating Conditions → Perceived Usefulness of BIS -> BIS adoption	0.023	0.039	0.602	0.547
Facilitating Conditions → Perceived Ease of Adoption of BIS -> BIS adoption	-0.007	0.036	0.188	0.851
Social Influence → Perceived Ease of Adoption of BIS -> BIS adoption	0.349	0.040	8.636	0.000
Information Quality → Perceived Ease of Adoption of BIS -> BIS adoption	0.281	0.056	4.984	0.000
Information Quality → Perceived Usefulness of BIS -> BIS adoption	0.005	0.009	0.576	0.565
Social Influence → Perceived Usefulness of BIS -> BIS adoption	0.005	0.009	0.544	0.587

Note: * p<0.01(1.96)

SMEs that can implement robust information technologies can now operate more efficiently and solve supply chain problems. However, the decision to adopt technologies and innovations to improve the inventory processes mentioned includes different edges associated with technological, organizational and environmental factors is called the TOE model, which is a transversal conceptual research model proposed by [10]. This theoretical model has been widely used in information systems to study the influences of technological, organizational and environmental contexts on SMEs' decision-making. Therefore, the TOE model can help explore the effects of current digital technologies and the environment on the diversification of SMEs' activities.

A previous study by [8] extended TAM in project communications and training and shared beliefs. Their findings suggest that training and communication in project communication influence TAM since perceived usefulness and perceived ease of use contribute to behavioral intention to use the technology. This leads to the impact of project management on the perceived ease of implementation. The author in [73] suggested successful business process change, user support and involvement, and vendors' expertise as critical success factors in ERP projects. The authors in [64] as well as [120] suggested that managing organizational change is critical for successful IT project implementation. The authors in [6] and [19] concluded that knowledge sharing enables a project team to reduce rework and compresses the time it takes to plan projects. They also stated that having the "right knowledge" to the "right person(s)" at the "right time" allows for greater control over the project throughout the project's lifecycle by reducing uncertainty. Therefore, the project management is driven by the change management and knowledge sharing in companies.

The mediation effect test was also employed in the current study. This study has found that perceived usefulness of BIS doesn't mediate the relationship between facilitating conditions and BIS adoption, perceived ease of adoption of BIS doesn't mediate the relationship between facilitating conditions and BIS adoption, while perceived ease of adoption of BIS mediates the relationship between social influence and BIS adoption with beta = 0.349, T-value = 8.636, and P-value = 0.000. Perceived ease of adoption of BIS mediates the relationship between information quality and BIS adoption with beta = 0.281, T-value = 4.984, and P-value = 0.000. The perceived usefulness of BIS doesn't mediate the relationship between information quality and BIS adoption. Meanwhile, perceived usefulness of BIS doesn't mediate the relationship between social influence and BIS adoption.

These results differed from those of the results of previous studies, where several previous studies investigated factors that led to the implementation of BIS. These factors were varied due to the type of industry within the SME business environment; the study of [50] developed a conceptual model. The model suggested by [50] was developed based on three levels of order that led to the implementation. The first order level comprises five factors: quality of information, system performance, user information, user training, and top management support. In brief, this level stands on information, users, and management. While the second level order comprises three factors which are subjective norm, perceived usefulness, and perceived ease of use, which resulted from the first order level factors, as these factors are supported by the theoretical debate of the technology acceptance model.

[81, 49] found significant mediation of the perceived ease of implementation of BIS between information quality and BIS implementation. Also, [81, 38, 49] found significant mediation of the perceived ease of implementation of BIS between IT project management and the BIS implementation

IV. CONCLUSION

In conclusion, this study explored the impact of various factors, including facilitating conditions, information quality, perceived ease of adoption of Business Intelligence Systems (BIS), perceived usefulness of BIS, and social influence, on the adoption of BISs in Libyan Small and Medium Enterprises (SMEs). The findings of this study indicated that these factors play a significant role in influencing the adoption of BIS in Libyan SMEs. Facilitating conditions, such as the availability of necessary resources and supportive infrastructure, positively influenced BIS adoption. Furthermore, the quality of information was identified as an essential factor affecting the adoption process. The perceived ease of adoption of BIS and its perceived usefulness are important factors influencing the decision to adopt BIS in Libyan SMEs. The study also identified social influence, such as the influence of peers and industry experts, as a factor that can shape the adoption behavior of SMEs.

These findings provide valuable insights for Libyan SMEs and policymakers looking to enhance the adoption of BIS. Recognizing the importance of creating facilitating conditions by providing necessary resources and supportive infrastructure can facilitate successful BIS adoption. Improving information quality is crucial to enable SMEs to harness the full potential of BIS. Emphasizing the ease of adoption and highlighting the usefulness of BIS can also positively influence the decision-making process. Lastly, acknowledging the influence of social factors and leveraging peer networks and industry experts can help in promoting BIS adoption among Libyan SMEs. By considering and addressing these factors, Libyan SMEs can enhance their competitiveness, decision-making capabilities, and overall operational efficiency through the successful adoption and utilization of BISs.

The current research recommends performing a comparative analysis between different types of SMEs in Libya to explore variations in the impact of the identified factors on BIS adoption. This study can investigate whether factors differ based on industry sector, company size, or geographical location and how these variations influence the adoption of BIS in SMEs.

A. Contribution to Academic Literature

The current study contributes to academic literature in the area of the development of a new, customized model for BISs adoption in SMEs operating in politically unstable environments like Libya. The model integrates both social and technological perspectives, which is a novel approach in this research domain. Our study further extends the Technology Acceptance Model (TAM) and UTAUT Theory by incorporating unique factors such as "Change Management", "IT Project Management", "Social Influence", and "Information Quality", specifically tailored for developing economies. Additionally, we utilize Structural Equation Modeling (SEM) to validate and explore relationships among various determinants of BIS adoption, thereby enhancing methodological rigor. This was made possible by providing a comprehensive conceptual framework linking technological, psychological, and organizational factors, which future researchers can build upon in similar contexts. These become an academic novelty when addressing BIS adoption within conflict-affected environments and serve as a significantly underexplored context in existing literature.

B. Contribution to Industry

The current study delivers a practical and implementable model to help SME managers effectively adopt BIS in alignment with their operational and technological constraints. The model is designed to fit the limited capabilities and resources typical of SMEs in Libya, making it a relevant tool for improving operational efficiency and informed decision-making. Our study further emphasizes the critical role of user perceptions (perceived ease and usefulness) in BIS adoption, guiding companies to develop more effective training and awareness programs for staff. Additionally, the present study offers practical guidance on enhancing information quality and knowledge-sharing practices, enabling SMEs to gain a competitive edge. All these industrial contributions are distinctive in that the solutions are custom-built for Libya's context, rather than adopting generalized models from developed countries.

C. Contributing to Policy

The present study supports policymakers in promoting digital transformation policies by encouraging BIS adoption among SMEs, based on concrete evidence provided in the research. The findings can guide public support programs for financing, training, and technical assistance tailored towards priority sectors. Our study can also be used to reinforce the importance of data-driven governance frameworks, using BIS to generate actionable performance metrics across industries. Advocates for government action to tackle infrastructure gaps, cultural barriers, and skills shortages, which were identified as major obstacles to effective BIS implementation. This can serve as a novel policy orientation for BIS adoption model grounded in the area of data from Libya, offering a clear roadmap for strategic public interventions.

V. Limitations of the Study

It is essential to acknowledge the limitations of the current study. The current research is limited to the following:

A. Sample Size and Representativeness

The study had a limited sample size, which restricted the generalizability of the findings. The findings may not reflect the perspectives and experiences of all Libyan SMEs. A larger and more diverse sample could provide a more comprehensive understanding of the adoption factors of BI systems in Libyan SMEs.

B. Data Collection Method

The study's findings are dependent on the data collection method employed, which is a survey. The limitations inherent in the chosen method, such as response bias or limited depth of information, might impact the study's outcomes. Using multiple data collection methods or employing a mixed-methods approach could enhance the robustness of the study.

C. Self-reporting Bias

The study was subject to self-reporting bias, as participants provided responses they perceived as socially desirable or were influenced by their own perceptions and biases. This bias could affect the accuracy and reliability of the data collected. Employing validation techniques or triangulating data from multiple sources helped mitigate this bias.

D. Timeframe and Context

The study's findings were time and context-specific, reflecting the conditions and circumstances prevalent during the study period. As technology and business environments evolve rapidly, the adoption factors of BI systems in Libyan SMEs might change over time. Therefore, the study's findings were interpreted within the specific timeframe and context in which they were conducted.

E. Causality and Correlation

The study established correlations between adoption factors and BI system adoption, but it did not establish a causal relationship. Other unmeasured or omitted variables could influence the relationship between adoption factors and BI adoption. Longitudinal studies or experimental designs could provide stronger evidence of causal relationships. Acknowledging these limitations is crucial in understanding the scope and applicability of the study's findings. Future research can address these limitations by expanding the sample size, utilizing diverse data collection methods, considering broader contextual factors, validating data through multiple sources, conducting longitudinal studies, and employing experimental designs to establish causality

References

[1] Abu-Shanab, Emad A. 2017. E-government familiarity influence on Jordanians' perceptions. Telematics and Informatics 34(1): 103-113.

[2] Agarwal, A. 2023. AI adoption by human resource management: a study of its antecedents and impact on HR system effectiveness. Foresight 25(1): 67-81.

[3] Ahmad, K., Ayyash, M. M., & Al-Qudah, O. M. A. (2018). The effect of culture values on consumer intention to use Arabic e-commerce websites in Jordan: an empirical investigation. *International Journal of Business Information Systems*, *29*(2), 155-182. https://doi.org/10.1504/IJBIS.2018.094691

[4] al-Eisawi, D., Serrano, A. & Koulouri, T. 2021. The effect of organisational absorptive capacity on business intelligence systems efficiency and organisational efficiency. Industrial Management and Data Systems 121(2): 519-544.

[5] al-Emran, Mostafa, Vitaliy Mezhuyev & Adzhar Kamaludin. 2018. Technology acceptance model in m-learning context: a systematic review. Computers and Education 125: 389-412.

[6] Alhadi Abu Baker Alhadi Almbrok & Abdullah Mohamed Ahmed Ayedh. 2017. Determinants of the access to sources of finance for Small and Medium Enterprises (SMES) in Libya: a conceptual study. International Journal of Social Sciences 1(1): 1040-1055.

[7] Al-Okaily, A., Teoh, A. P. & al-Okaily, M. 2023. Evaluation of data analytics-oriented business intelligence technology effectiveness: an enterprise-level analysis. Business Process Management Journal 29(3): 777-800.

[8] al-Zayyat, Abdel Naser, al-Khaldi, Firas, Ibrahem Tadros & al-Edwan, Ghassan. 2010. The effect of knowledge management processes on project management. IBIMA Business Review 3: 1-6.

[9] Ameen, A. A., & Ahmad, K. (2017, 2017//). Information Systems Strategies to Reduce Financial Corruption. (Ed.),^(Eds.). Leadership, Innovation and Entrepreneurship as Driving Forces of the Global Economy, Cham.

[10] Amoako-Gyampah, K & Salam, A. F. 2004. An extension of the technology acceptance model in an ERP implementation environment. Information and Management 41: 731-745.

[11] Arnaboldi, M., Robbiani, A. & Carlucci, P. 2021. On the relevance of self-service business intelligence to university management. Journal of Accounting and Organizational Change 17(1): 5-22.

[12] Arora, M. & Sharma, R. L. 2023. Artificial intelligence and big data: ontological and communicative perspectives in multi-sectoral scenarios of modern businesses. Foresight 25(1): 126-143.

[13] Bach, M. P., Zoroja, J. & Čeljo, A. 2017. An extension of the technology acceptance model for business intelligence systems: project management maturity perspective. International Journal of Information Systems in the Service Sector 5(2): 5-21.

[14] Bahar, N., Wahab, S. N., & Rahman, M. (2021). Impact of knowledge management capability on supply chain management and organizational practices in logistics industry. *VINE Journal of Information and Knowledge Management Systems*, *51*(5), 677-692. https://doi.org/10.1108/VJIKMS-02-2020-0022

[15] Batara, E., Nurmandi, A. Warsito, T. & Pribadi, U. 2017. Are government employees adopting local e-government transformation? Transforming Government: People, Process and Policy 11(4): 612-638.

[16] Bhatt, P. 2023. AI adoption in the hiring process-important criteria and extent of AI adoption. Foresight 25(1): 144-163.

[17] Božič, K. & Dimovski, V. 2019. Business intelligence and analytics for value creation: the role of absorptive capacity. International Journal of Information Management 46(1): 93-103.

[18] Caseiro, N. & Coelho, A. 2018. Business intelligence and competitiveness: the mediating role of entrepreneurial orientation. Competitiveness Review: An International Business Journal 28(2): 213-226.

[19] Caseiro, N. & Coelho, A. 2019. The influence of business intelligence capacity, network learning and innovativeness on startups performance. Journal of Innovation and Knowledge 4(3): 139-145.

[20] Chaudhuri, R., Chatterjee, S., Kraus, S. & Vrontis, D. 2023. Assessing the AI-CRM technology capability for sustaining family businesses in times of crisis: the moderating role of strategic intent. Journal of Family Business Management 13(1): 46-67.

[21] Chawla, D. & Joshi, H. 2020. The moderating role of gender and age in the adoption of mobile wallet. Foresight 22: 483-504.

[22] Chen, L., Jiang, M., Jia, F. & Liu, G. 2022. Artificial intelligence adoption in business-to-business marketing: toward a conceptual framework. Journal of Business and Industrial Marketing 37(5): 1025-1044.

[23] Chen, Y., Hu, Y., Zhou, S. & Yang, S. 2023. Investigating the determinants of performance of artificial intelligence adoption in hospitality industry during COVID-19. International Journal of Contemporary Hospitality Management 35(8): 2868-2889.

[24] Cheng-Che Shen, Ray- E Chang, Ching Jou Hsu & I-Chiu Chang. 2017. How business intelligence maturity enabling hospital agility. Telematics and Informatics 34(1): 450-456.

[25] Creswell, J. W. & Creswell, J. D. 2017. Research Design: Qualitative, Quantitative and Mixed Methods Approaches. s.l: Sage publications.

[26] Dai, H., Wen, Y., Zhou, W., Tong, T. & Xu, X. 2023. Enhancing online-to-offline delivery efficiency facilitated by Industry 4.0: a personnel configuration perspective. Industrial Management and Data Systems 123(4): 1198-1219.

[27] Davis, F. D. & Venkatesh, V. 1996. A critical assessment of potential measurement biases in the technology acceptance model: three experiments. International Journal of Human-Computer Studies 45(1): 19-45.

[28] Davis, F. D. 1986. A Technology Acceptance Model for Empirically Testing New End-User Information Systems. Cambridge: s.n.

[29] Davis, F. D. 1989. Perceived usefulness, perceived ease of use and user acceptance of information technology. MIS Quarterly 13(3): 319-340.

[30] Davis, F. D., Bagozzi, R. P. & Warshaw, P. R. 1989. User acceptance of computer technology: a comparison of two theoretical models. Management Science 35(8): 982-1003.

[31] Davis, F. D., Bagozzi, R. P. & Warshaw, P. R. 1992. Extrinsic and intrinsic motivation to use computers in the workplace. Journal of Applied Social Psychology 22(14): 1111-1132.

[32] Dresner, H. 1989. Business Intelligence. Slide. Gartner Inc.

[33] Fang, M., Liu, F., Xiao, S. & Park, K. 2023. Hedging the bet on digital transformation in strategic supply chain management: a theoretical integration and an empirical test. International Journal of Physical Distribution and Logistics Management 53(4): 512-531.

[34] Fishbein, M. & Ajzen, I. 1975. Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research. s.l.: Addison-Wesley.

[35] Flavián, C., Pérez-Rueda, A., Belanche, D. & Casaló, L. V. 2022. Intention to use analytical artificial intelligence (AI) in services-the effect of technology readiness and awareness. Journal of Service Management 33(2): 293-320.

[36] Foshay, N., Taylor & Mukherjee, A. 2014. Winning the hearts and minds of business intelligence users: the role of metadata. Information Systems Management 31(2): 167-180.

[37] Fu, H. P., Chang, T. H., Lin, S. W., Teng, Y. H. & Huang, Y. Z. 2023. Evaluation and adoption of artificial intelligence in the retail industry. International Journal of Retail and Distribution Management 51(6): 773-790.

[38] Garg, R. 2016. Methodology for research I. Indian Journal of Anaesthesia 60(9): 640-645.

[39] Gatignon, H. & Xuereb, J. M. 1997. Strategic orientation of the firm and new product performance. Journal of Marketing Research 34(1): 77-90.

[40] Ghobakhloo, M., Iranmanesh, M., Vilkas, M., Grybauskas, A. & Amran, A. 2022. Drivers and barriers of Industry 4.0 technology adoption among manufacturing SMEs: a systematic review and transformation roadmap. Journal of Manufacturing Technology Management 33(6): 1029-1058.

[41] Gina, B. & Budree, A. 2020. A review of literature on critical factors that drive the selection of business intelligence tools. 2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), pp. 1-7.

[42] Goel, P., Kaushik, N., Sivathanu, B., Pillai, R. & Vikas, J. 2022. Consumers' adoption of artificial intelligence and robotics in hospitality and tourism sector: literature review and future research agenda. Tourism Review 77(4): 1081-1096.

[43] Gonzales, R. & Wareham, J. 2019. Analysing the impact of a business intelligence system and new conceptualizations of system use. Journal of Economics, Finance and Administrative Science 24: 345-368.

[44] Haggenmüller, S., Oehlschläger, P., Herbst, U. & Voeth, M. 2023. Time for change? scenario analysis on buyer-seller negotiations. Journal of Business and Industrial Marketing 38(5): 1215-1242.

[45] Hamidinava, F., Ebrahimy, A., Samiee, R. & Didehkhani, H. 2023. A model of business intelligence on cloud for managing SMEs in COVID-19 pandemic. Kybernetes 52(1): 207-234.

[46] Hao, S. & Song, M. 2016. Technology-driven strategy and firm performance: are strategic capabilities missing links?. Journal of Business Research 69(2): 751-759.

[47] Hoang, T. G. & Bui, M. L. 2023. Business intelligence and analytic (BIA) stage-of-practice in micro-, small- and medium-sized enterprises (MSMEs). Journal of Enterprise Information Management 36(4): 1080-1104.

[48] Jaklič, J., Grublješič, T. & Popovič, A. 2018. The role of compatibility in predicting business intelligence and analytics use intentions. International Journal of Information Management 43: 305-318.

[49] Jee-Won Kang & Young Namkung. 2019. The information quality and source credibility matter in customers' evaluation toward food O2O commerce. International Journal of Hospitality Management 78: 189-198.

[50] Johl, S. K., Ali, K., Shirahada, K., & Oyewale, O. I. (2024). Green servitization, circular economy, and sustainability a winning combination analysis through hybrid SEM-ANN approach. *Business Strategy and the Environment*, *33*(8), 8978-8993.

[51] Kabra, G. 2023. Determinants of blockchain adoption and organizational performance in the healthcare sector in India. American Journal of Business 38(3): 152-171.

[52] Kapetaneas, N. & Kitsios, F. 2022. Digital transformation approach to public hospitals environment: technology acceptance model for business intelligence applications. In Themistocleous, M. & Papadaki, M. (ed.). Information Systems, pp. 453-62. Cham: Springer International Publishing.

[53] Kohnke, O., Wolf, T. R. & Mueller, K. 2011. Managing user acceptance: an empirical investigation in the context of business intelligence standard software. International Journal of Information Systems Change Management 5(4): 269-290.

[54] Kostelník, P. & Dařena, F. 2022. Conversational interfaces for unconventional access to business relational data structures. Data Technologies and Applications 56(1): 87-102.

[55] Koul, S. & Eydgahi, A. 2017. A systematic review of technology adoption models and their applications. Journal of technology management and innovation 12(4): 106-113.

[56] Krejcie, R. & Morgan, D. 1970. Determining sample size for research activities. Educational and Psychological Measurement 30: 607-610.

[57] Kuei-Fang Ho, Cheng-Hsun Ho & Min-Huey Chung. 2019. Theoretical integration of user satisfaction and technology acceptance of the nursing process information system. PloS One 14(6): 1-14.

[58] Kumari, S., Raghuram, P., Venkatesh, V. G. & Shi, Y. 2022. Future perspectives on progressive farming with adoption of virtual reality technology for sustainable quality in agriculture. The TQM Journal 34(2): 250-279.

[59] Kwonsang Sohn & Ohbyung Kwon. 2020. Technology acceptance theories and factors influencing artificial Intelligence-based intelligent products. Telematics and Informatics 47.

[60] Lee, J. C. & Chen, X. 2022. Exploring users' adoption intentions in the evolution of artificial intelligence mobile banking applications: the intelligent and anthropomorphic perspectives. International Journal of Bank Marketing 40(4): 631-658.

[61] Lee, J.C. & Lin, R. 2023. The continuous usage of artificial intelligence (AI)-powered mobile fitness applications: the goal-setting theory perspective. Industrial Management and Data Systems 123(6): 1840-1860.

[62] Leite, N., Pedrosa, I. & Bernardino, J. 2018. Comparative evaluation of open source business intelligence platforms for SME. 2018 13th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1-6.

[63] Leite, N., Pedrosa, I. & Bernardino, J. 2019. Open source business intelligence on a SME: a case study using Pentaho. 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1-7.

[64] Liu, W., Wang, J., Jia, F. & Choi, T.M. 2022. Blockchain announcements and stock value: a technology management perspective. International Journal of Operations and Production Management 42(5): 713-742.

[65] Luhn, H. P. 1958a. A Business Intelligence System. IBM Journal of Research and Development 2(4): 314-319.

[66] Luhn, H. P. 1958b. The automatic creation of literature abstracts. IBM Journal of Research and Development 2(2): 159-165.

[67] Markus, M. L. 2004. Technochange management: using IT to drive organizational change. Journal of Information Technol ogy 19: 4-20.

[68] Maya Fouad Farah, Muhammad Junaid Shahid Hasni & Abbas Khan Abbas. 2018. Mobile-banking adoption: empirical evidence from the banking sector in Pakistan. International Journal of Bank Marketing 36(1): 1386-1413.

[69] Mohammad Daradkeh & al-Dwairi, Radwan Moh'd. 2018. Self-service business intelligence adoption in business enterprises: the effects of information quality, system quality and analysis quality. International Journal of Enterprise Information System 13(3): 65-85.

[70] Muhammad Bakhsh, Amjad Mahmood & Nazir A. Sangi. 2017. Examination of factors influencing students and faculty behavior towards m-learning acceptance. The International Journal of Information and Learning Technology 34(3): 166-188.

[71] Narteh, B., Mahmoud, M. A. & Amoh, S. 2017. Customer behavioural intentions towards mobile money services adoption in Ghana. The Service Industries Journal 37: 426-447.

[72] Naseri, R. N. N., Yusuf, A. H., & Oyelakin, I. O. (2023). Determinant of purchase intention through online on Halal cosmetic. *Malaysian Journal of Consumer and Family Economics*, *31*(1).

[73] Nath, S. D., Khayer, A., Majumder, J. & Barua, S. 2022. Factors affecting blockchain adoption in apparel supply chains: does sustainability-oriented supplier development play a moderating role?. Industrial Management and Data Systems 122(5): 1183-1214.

[74] Nayal, K., Raut, R., Priyadarshinee, P., Narkhede, B. E., Kazancoglu, Y. & Narwane, V. 2022. Exploring the role of artificial intelligence in managing agricultural supply chain risk to counter the impacts of the COVID-19 pandemic. The International Journal of Logistics Management 33(3): 744-772.

[75] Ngai, E. W. T., Law, C. C. H. & Wat, F. K. T. 2008. Examining the critical success factors in the adoption of enterprise resource planning. Computers in Industry 59(6): 548-564.

[76] Noor Ul-Ain, Giovanni, V., DeLone, W. H. & Mehwish Waheed. 2019. Two decades of research on business intelligence system adoption, utilization and success - a systematic literature review. Decision Support Systems 125: 1-13.

[77] Noor Ul-Ain, Giovanni, V., DeLone, W. H. 2019. Business intelligence system adoption, utilization and success - a systematic literature review. Proceedings of the 52nd Hawaii International Conference on System Sciences, pp. 5888-5897.

[78] Okunlaya, R. O., Syed Abdullah, N. & Alias, R. A. 2022. Artificial intelligence (AI) library services innovative conceptual framework for the digital transformation of university education. Library Hi Tech 40(6): 1869-1892.

[79] Oyelakin, I. O., Ting, D. H., Yusuf, A. H., Arbak, S., & Dhar, B. K. (2025). Building Resource Capabilities Through Green Servitization and ISO 14001 for Sustainable Performance: Perspectives From Manufacturing Firms. *Corporate Social Responsibility and Environmental Management*.

[80] Oyewale, O. I. (2023). *Examining Causal Relationship of Servitization, and Green Servitization on Sustainable Performance: Mediating Role of ISO 14001 on Malaysian Public Listed Manufacturing Firms* (Doctoral dissertation, Universiti Teknologi PETRONAS)

[81] Roslan, m. r. h., ahmad, k., & ayyash, m. m. (2020). Factors influencing information systems quality from the system developers perspective. *Asia-Pacific Journal of Information Technology and Multimedia*, *9*(1), 82-93. https://doi.org/dx.doi.org/10.17576/apjitm-2020-0901-07

[82] Rosman, m. r. m. (2020). The role of business processes in influencing the decision support capabilities of Enterprise Content Management System (Ecms): Towards a framework. *Asia-Pacific Journal of Information Technology and Multimedia*, *9*(1), 58-68. https://doi.org/dx.doi.org/10.17576/apjitm-2020-0901-05

[83] Yahaya, J. H., Abai, N. H. Z., Deraman, A., Hamdan, A. R., Mansor, Z., & Jusoh, Y. Y. (2018). The Integrated Framework of Business Intelligence and Data Analytics for Organizational Performance Management in Public Sector. (Ed.),^(Eds.). Proceedings of the World Congress on Engineering.

[84] Yusof, M. M. (2019). A socio-technical and lean approach towards a framework for health information systems-induced error (*Improving Usability, Safety and Patient Outcomes with Health Information Technology* (pp. 508-512). IOS Press. https://doi.org/doi:10.3233/978-1-61499-951-5-508

[85] Yusof, M. M., & Aziz, K. (2015). Evaluation of organizational readiness in information systems adoption: A case study. *Asia-Pacific Journal of Information Technology Multimedia*, *4*(2), 69-86. https://doi.org/dx.doi.org/10.17576/apjitm-2015-0402-06

An Improved BFT Algorithm in Traceability Data for Supply Chain

Zhiyong Liang[1]*, Rongwang Jiang[2], Ming Yang[3], Boxiong Yang[4]
School of Information and Intelligence Engineering, University of Sanya, Sanya 572022 [1, 2, 3, 4]
Supercomputing Center, University of Sanya, Sanya 572022 [1, 4]
Academician Workstation of Chunming Rong, University of Sanya, Sanya 572022 [1, 2, 3]
Academician Workstation of Guoliang Chen, University of Sanya, Sanya 572022 [4]

***Abstract*—Byzantine Fault Tolerance (BFT) is a class of fault-tolerance techniques in the field of distributed computing. Aiming at the risks of error-prone and tampering data brought by the centralized database in the traditional supply chain traceability process, the use of the BFT consensus algorithm in combination with the alliance blockchain can be used to solve the security problems such as data deletion, data misuse, application attacks, and efficiency reduction in the storage process of supply chain traceability data. This will be the future trend of safe and orderly storage and management of supply chain traceability data.**

***Keywords*—Tendermint; BFT; consensus; consortium blockchain; traceability data for supply chain**

I. Introduction

Tendermint began in early 2014 as an improved Byzantine Fault Tolerance (BFT), born out of the open-source realization that while the PoW was secure in the network of BTC, it had serious shortcomings in terms of speed and scalability [1-5]. BFT is designed to handle these abnormal behaviors and to meet the specification requirements of the problem to be solved. Tendermint is a secure state machine replication algorithm in the blockchain paradigm, in the form of BFT-ABC with additional accountability to facilitate verification of Byzantine node dishonesty [6,7]. The core of Tendermint runs on all nodes and handles consensus, P2P networking, blockchain storage and interaction with the outside via RPC functions. Each Tendermint core can talk to a custom blockchain application, a separate process, through a network socket protocol called TMSP [8].

A. Consensuses

Consensus is used to solve the problem of consistency in distributed systems and is centered on making a specified operation consistent, recognized, and tamper-proof across a distributed network for a limited period of time, guaranteed by some protocol. In blockchain systems, specific consensus is used to solve the problem of decentralized multi-party mutual trust, while each node must make its own ledger consistent with that of other nodes. And in a centralized world, this is almost impossible due to the presence of a centralized server. Currently, the more common consensuses in blockchain are PoW, PoS, DPoS, Raft, Ripple, PBFT, etc. [9, 10].

B. PoW (Proof of Work)

Most public chains or digital virtual currencies, such as BTC, ETH, ETC, etc. use PoW for implementing their consensus mechanism, which determines the distribution of the currency based on the effective work contributed by mining [11, 12]. The consistency problem of decentralized bookkeeping systems is solved in the network of BTC by competing for bookkeeping, i.e. competing for bookkeeping rights through hashrate [13, 14]. ETH, hashrate is no longer handled by mere CPUs, and it is now common to use GPUs, FPGAs and ASICs for mining, which directly leads to the gradual centralization of hashrate. From a long-term development perspective, PoW runs counter to decentralization, and the security of blockchain networks is threatened [15, 16].

C. PoS (Proof of Stake)

In [17, 18], PoS is similar to property being stored in a bank, and this model allocates interest to the holder based on the amount and length of time the holder has held the digital currency. Peercoin is the first digital cryptocurrency to use proof-of-stake, which introduces the concept of coin-age in terms of the difficulty of the hash operation of SHA256, making the difficulty inversely proportional to the coin-age of the transaction input. While PoS takes into account the shortcomings of PoW, it is based on the fact that equity balances can lead to greater power in the richest accounts and potentially dominate bookkeeping rights. POS also has the following four potential problems: Nothing at Stake Problem, Initial Distribution Problem, Long-Range Attack and Coin Age Accumulation Attack.

D. DPoS (Delegated Proof of Stake)

In [19, 20] The emergence of DPoS addresses some of the shortcomings in PoW and PoS. Bitshare is a class of digital cryptocurrencies using the DPoS mechanism, which expects to reduce the negative effects of centralization by introducing a technical democratic layer. The risk of DPOS is that there are many difficulties in dealing with faulty nodes, and community elections cannot effectively stop some sabotage nodes in time, creating a security risk for the network. At the same time, if the elected super node does not have strong computing power to protect itself, it is easy to be attacked by DDOS, which will seriously affect the network stability.

E. Raft

In distributed scenarios where the assumptions are made that the conditions do not require Byzantine nodes to be considered, but only deal with general dead failures, it is generally more efficient to use a protocol, such as Paxos [21, 22]. Compared to Paxos, Raft has essentially the same operational efficiency, but

*Corresponding author.

is easier to understand and more likely to be used in system development. Although Raft is a very efficient consensus algorithm, it can only support CFT (Crash Fault Tolerance), which only allows nodes in the system to crash, and does not consider malicious nodes in the system, so Raft is only applicable to Private Blockchain.

F. Ripple

In [23-26], Ripple is an Internet-based open-source payment protocol that enables decentralized cryptographic digital currency exchange, payment and clearing functions. In the network of Ripple, transactions are initiated by the client and are broadcast throughout the network via tracking nodes or validating nodes. Ripple is more efficient than anonymous consensuses such as PoW, with transactions taking only a few seconds to confirm. However, this advantage also dictates that this consensus is only suitable for use in Permissioned Chains. Ripple has a BFT capability of (n-1)/5, which can tolerate Byzantine errors in 20% of the nodes in the entire network without affecting the correct consensus. Compared to BFT, Ripple tolerates a low percentage of Byzantine nodes.

G. PBFT (Practical Byzantine Fault Tolerance)

In Consortium Blockchain, most of them use consensuses such as PBFT, Raft, Paxos, etc. This type of consensus is a strong consistency algorithm and has a huge advantage in consensus speed. PBFT is a more mature and widely used consensus, and is a widely used message-passing based consistency algorithm. First, PBFT overcomes the inefficiency of the original BFT and enhances the operability in practical applications of distributed systems. Second, compared to PoW and PoS, which are often used in Public blockchain, PBFT does not require nodes to consume hashrate to solve the meaningless proof-of-work problem and thus obtain the right to bookkeeping. Therefore, a blockchain system using PBFT as the consensus algorithm can effectively avoid the problem of monetary security caused by issuing digital currency, and avoid the high latency, low throughput, and meaningless consumption of hashrate in the consensus process.

PBFT addresses the problem of exponential increase of algorithm complexity with increasing nodes in BFT, and optimizes it to reduce the complexity of the algorithm from exponential level to polynomial level, making it possible to apply Byzantine protocol in distributed systems. PBFT solves the problem of inefficiency of the original Byzantine fault-tolerant algorithm and is applied to transactions performed in the presence of trust between parties. Fault tolerance means that with the use of this consensus algorithm, the entire coalition chain network can accommodate a certain number of Byzantine nodes and allow them not to affect the proper reaching of the consensus. PBFT can tolerate Byzantine errors in a third of the nodes in the entire network without affecting the correct consensus. PBFT is a Byzantine fault-tolerant state machine replication that uses one or more rounds of voting to obtain a majority consensus, and its flowchart is shown in Fig. 1, we can see that the innovation of PBFT lies in the design of three-stage voting, which is divided into three stages, namely Pre-prepare, Prepare, and Commit. 0, 1, 2, 3 represent the number of the node respectively, where 0 is the primary node primary, 3 represents the Byzantine node, where the behavior of the performance is no response to the requests of other nodes.

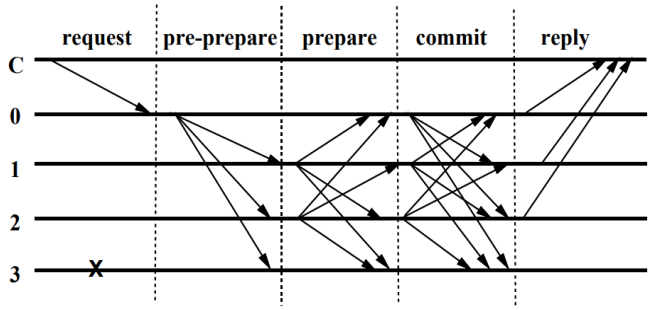


Fig. 1. Flow chart of PBFT.

At the same time, we can also see from Fig. 1 that PBFT requires three rounds of voting to achieve the consensus of the whole distributed system, but in the actual use of Consortium Blockchain, when the number of participating nodes is too many, the existence of the three-round voting mechanism makes the communication complexity still too high, which leads to a very rapid decline in the performance of the whole network, and the network latency also becomes high or even timeout.

II. TENDERMINT

A. Overview

Tendermint belongs to the BFT mechanism, an easy-to-understand and easy-to-implement State Machine Replication, which is based on the assumption of a semi-synchronous network that can tolerate 1/3 of Byzantine nodes with guaranteed Liveness and Safety. Tendermint is a Round-Based protocol that has been further optimized for PBFT. Compared to traditional PBFT, Tendermint ideally requires only two rounds of voting to reach consensus, which ensures that the entire network can reach consensus in a secure and reliable manner.

Tendermint is characterized by the following seven points:

1) Byzantine fault-tolerant: Tendermint tolerates up to 1/3 of your machines failing arbitrarily. This includes explicitly malicious behavior.

2) State machine replication: Tendermint can replicate deterministic state machines written in any programming language.

3) Ultimate certainty: Once all nodes reach consensus, Tendermint will not fail at some point in the future like the consensus of BTC or ETH, so there will be no blockchain fork in Tendermint.

4) Secure P2P: Gossip protocols and peer discovery are secured via Tendermint's authenticated encryption system.

5) Lightning fast: By avoiding POW consensus, Tendermint can achieve very high transaction throughput. With ideal application data structure support, 42,000 transactions per second can be achieved. Tendermint supports thousands of transactions per second.

6) Easy to deploy: Use mintnet to deploy your app to any cloud provider supported by docker-machine.

7) 100% Open source: All Tendermint's source code is licensed under Apache 2.0.

B. Workflow

In the network of Consortium Blockchain based on BFP, nodes in the network can reach consensus when there are no more than 1/3 of the total number of Byzantine nodes. Tendermint is an easy-to-understand BFT consensus protocol that follows a simple state machine with the workflow diagram shown in Fig. 2.

In Fig. 2, there are two roles in Tendermint: Validator and Proposer. Validator is a role or node in the protocol, and different validators have different Vote Power in the voting process. The Proposer is elected by the Validator on a rotating basis, who takes turns proposing blocks for transactions and voting on proposed blocks. Blocks are committed to the chain and each block is a block height. However, it is possible for a block to fail to be submitted, in which case the protocol will select the next Proposer to propose a new block at the same height and restart the poll.

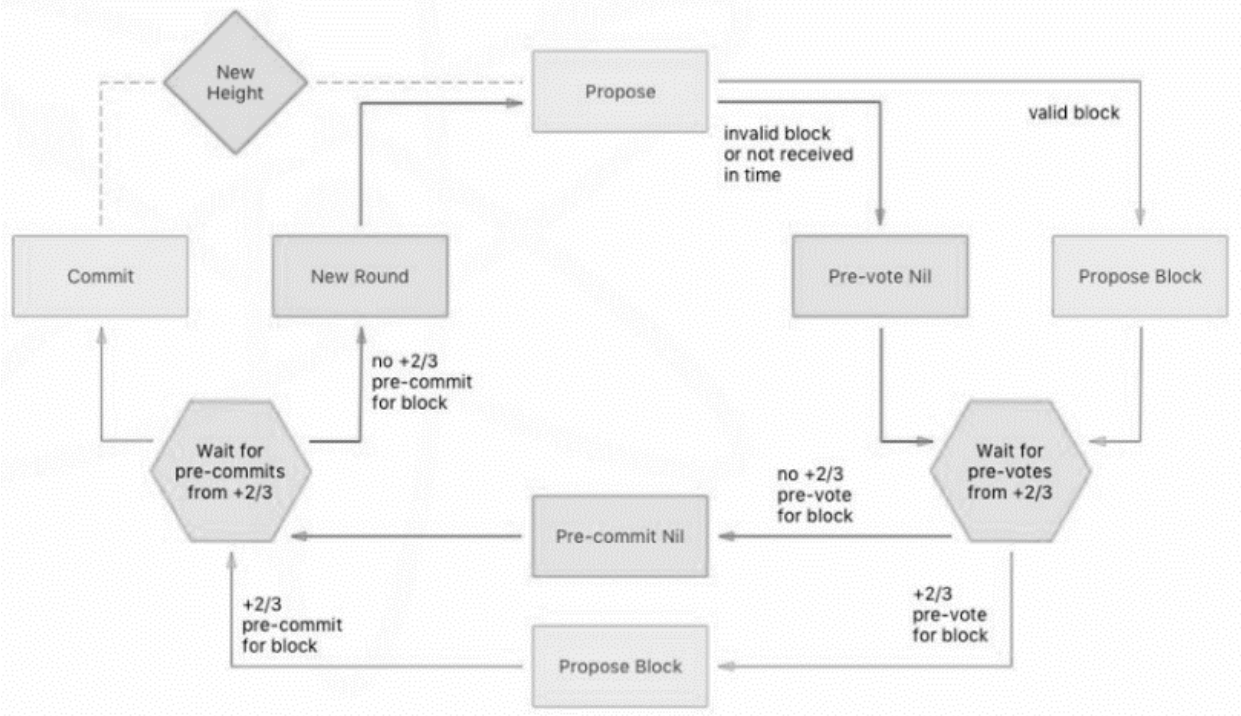


Fig. 2. Flow chart of Tendermint.

If a block needs to be successfully submitted, it must go through two stages of voting, called Pre-vote and Pre-commit. A block is committed only when more than 2/3 of Validators have Pre-committed votes on the same block in the same round of proposals. Proposers may fail to propose blocks due to reasons such as offline or network latency. This situation is also allowed in Tendermint, as Proposers will wait for a certain amount of time before moving to the next round of offers for receiving blocks proposed by Proposers.

If less than one-third of Validators are Byzantine nodes, Tendermint can guarantee that Validators will never commit blocks repeatedly at the same height and cause conflicts. To do this, Tendermint introduces a locking mechanism where once Validators have pre-polled a block, then that Validator will be locked to that block. If the block is not successfully submitted in the current round of pre-proposals and pre-votes, the Validator is unlocked and the next round of pre-submissions for the new block is performed.

C. Comparison Between Tendermint and PBFT

As can be seen from the workflow of both, Tendermint is more like an upgraded and optimized version of PBFT, and the similarities and differences between the two are as follows.

1) Similarities

- Both belong to the BFT system.
- All are resistant to 1/3 Byzantine node attacks.
- Three-stage submission, with the first stage broadcasting transactions (blocks) and the last two stages broadcasting signatures (confirmations).
- Both require a quorum to submit a block.

2) Differences

- The difference between Tendermint and PBFT is mainly that the two make different judgments when more than 1/3 of the Byzantine nodes are network-wide nodes. PBFT can't provide guarantees when the number of Byzantine nodes is between 1/3 and 2/3 of the number of Validators, allowing the attacker to return arbitrary results to the client. And Tendermint considers that more than 2/3 number of pre-commit acknowledgements are required to commit a block. For example, if 1/2 of the Validators are Byzantine nodes, in Tendermint these Byzantine nodes can block the submission of blocks, but

they themselves cannot submit malicious blocks, but in PBFT Byzantine nodes are able to submit malicious blocks to the client.

- In the Byzantine node concept, PBFT refers to the number of nodes, while Tendermint represents the number of interests, or voting power, of the nodes.
- While PBFT requires a pre-defined set of Validators for the validation process, Tendermint supports dynamic changes in Validators by requiring more than 2/3 of the quorum of validators to approve membership changes.

III. Consensus Performance in Traceability Data for Supply Chain Based on Tendermint and PBFT

A. Problems of Current Supply Chain Traceability Data

In response to the problems such as serious centralization, easily tampered information and opaque information in the process of traceability of supply chain data, Consortium Blockchain can be used to realize the safe and effective storage of traceability data. Although PBFT is currently more widely used in federated chains, the theoretical comparison in Chapter 3 shows that using Tendermint is superior to PBFT in terms of performance and speed.

B. Experimental Environment Setup and Preparation

The experimental part of this paper runs on the node1 in Supercomputing Center, University of Sanya, which has 20 Intel(R) Xeon(R) Silver 4114 CPUs @ 2.20GHz, 128G of RAM, and 436T of hard disk. The operating system is CentOS Linux release 7.6.2009, and the Fabric-Consortium Blockchain is built by deploying k8s Master node and multiple k8s nodes with docker, where each k8s node represents a Consortium Blockchain deployment node. Since Fabric adopts a loosely coupled design, the components such as Consensus and identity verification are modularized to make it easy to select the corresponding modules according to the application scenarios during the application process. Therefore, Tendermint and PBFT will be used to test the consensus performance of supply chain data for comparison throughout the system respectively. During the test, the following five performance indicators were examined.

- Fail Rate: Percentage of operation failures (including timeouts)
- Avg Latency: Average time taken to complete a transaction
- Query Throughput: Query requests per second
- Consensus Throughput: Consensus requests per second
- Consistency Throughput: Synchronous operations per second

C. Results Without Byzantine Failures

PBFT was used to test the traceability data for supply chain, and the indicators are shown in Table I.

Tendermint was used to test the traceability data for supply chain, and the indicators are shown in Table II.

TABLE I. Table of Indicators Using PBFT

Serial Number	Test Contents	Data Quantity	Fail Rate	Avg Latency	Query Throughput	Consensus Throughput	Consistency Throughput
1	Supply Chain	1000	0	5.9s	948.3tps	577tps	525tps
2	Supply Chain	2000	0	15.24s	1874.2tps	570ps	530tps
3	Supply Chain	5000	0	42.09	4879.32tps	576tps	504tps

TABLE II. Table of Indicators Using Tendermint

Serial Number	Test Contents	Data Quantity	Fail Rate	Avg Latency	Query Throughput	Consensus Throughput	Consistency Throughput
1	Supply Chain	1000	0	1.32s	986.7tps	993tps	984tps
2	Supply Chain	2000	0	2.24s	1987.2tps	1991ps	1797tps
3	Supply Chain	5000	0	4.19s	4983.32tps	4985tps	1877tps

As we can see from Table III, after switching to Tendermint, the Consensus Throughput can reach up to 5000tps and the Consistency Throughput is close to 1900tps. At the same time, the average latency of Tendermint is also significantly reduced compared to PBFT. Such write performance is comparable to that of a traditional single-node relational database, and can satisfy most commercial scenarios.

From the data in we can see that the average latency of the algorithm is significantly lower than that of the PBFT consensus algorithm due to the further optimization of the algorithm and the reduction of the number of voting rounds of Tendermint, which reduces the complexity of the whole algorithm, and the comparison of the two latencies is shown in Fig. 3.

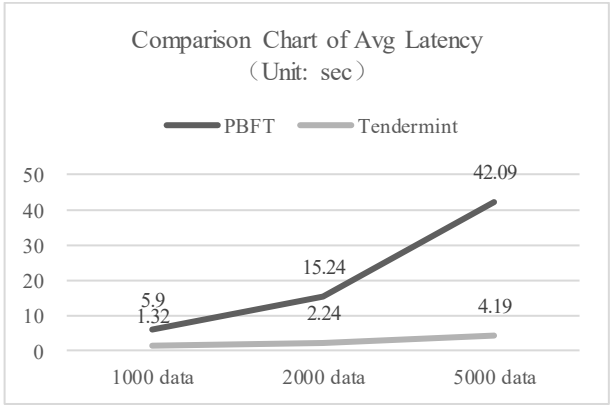


Fig. 3. Comparison chart of average latency.

For the efficiency of the two consensuses, the most critical metrics to examine are Query Throughput, Consensus Throughput, and Consistency Throughput. performance metrics comparison graph is shown in Fig. 4. From the figure, we can see that due to the adoption of the optimization algorithm, the data throughput of the whole model has been improved to some extent after using Tendermint, especially Consensus Throughput and Consistency Throughput, which are improved to a larger extent.

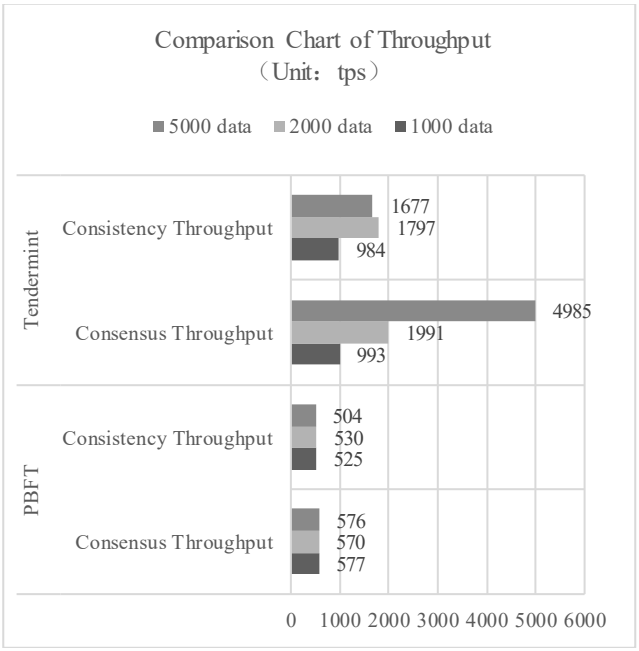
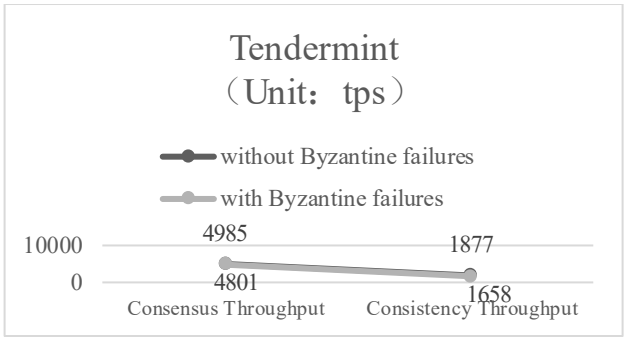


Fig. 4. Comparison chart of throughput.

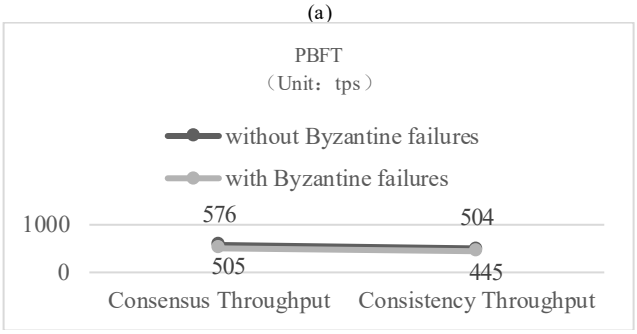
D. Results with Byzantine Failures

Despite the injected Byzantine faults, which would cause many systems to fail completely and immediately, Tendermint maintains respectable latencies. When processing more data and there are more Byzantine failures, Tendermint has lower latency and higher throughput than PBFT due to partial optimization in the voting node. The consensus performance of Tendermint and PBFT when processing 5000 produce traceability data and there are 30% Byzantine failures is shown in Table III.

As can be seen in Table III, with more traceability data and a higher percentage of Byzantine failures, Tendermint has a low performance loss of about 2%-4%, still maintaining a low Avg Latency and a high throughput. However, PBFT has a high-performance loss and excessive latency, which is shown in Fig. 5(a) and Fig. 5(b).



(a)



(b)

Fig. 5. (a) Core performance metrics comparison chart of Tendermint, (b) Core performance metrics comparison chart of PBFT.

TABLE III. CONSENSUS PERFORMANCE PARAMETERS WITH 30% BYZANTINE FAILURES

	Fail Rate	Avg Latency	Query Throughput	Consensus Throughput	Consistency Throughput
Tendermint	0%	5.38s	4791.9tps	4801tps	1658tps
PBFT	0%	50.56s	4364.63tps	505tps	445tps

IV. CONCLUSION

We introduced Tendermint, an improved blockchain consensus protocol based on BFT. Tendermint optimizes the election and voting sessions on the basis of PBFT and reduces the complexity so that it can have lower latency and better data throughput in the consensus process and is more excellent in resisting attacks with no more than 1/3 of Byzantine failures. In the experimental session, Tendermint outperforms PBFT.

FUNDING

This work is supported by Hainan Province Key R&D Program Project (Grant ZDYF2023GXJS007) and University of Sanya Major Special Projects (USY22XK-04). Meanwhile, the experimental part in this paper have been done on the supercomputing system in the Supercomputing Center, University of Sanya.

CONFLICT OF INTEREST

The authors declare that they have no conflicts of interest regarding this work.

DATA AVAILABILITY

All data generated or analyzed during this study are included in the manuscript.

CODE AVAILABILITY

Not applicable.

AUTHORS' CONTRIBUTIONS

Zhiyong Liang: Conceptualization, Methodology, Formal Analysis, Writing—Original Draft Rongwang Jiang (Rongwang Jiang): Supervision, Project Administration, Funding Acquisition, Writing—Review & Editing Ming Yang: Data Curation, Software, Validation, Visualization Boxiong Yang: Investigation, Resources, Writing—Review & Editing.

REFERENCES

[1] Z. Ge, D. Loghin, B. C. Ooi, *et al.*, "Hybrid blockchain database systems: Design and performance," *VLDB Endowment*, vol. 15, no. 5, pp. 1092–1104, 2022. doi: 10.14778/3514087.3514094.

[2] S. M. Alqahtani, "Analyzing and improving performance in BFT consensus protocols," unpublished.

[3] S. Alqahtani and M. Demirbas, "BigBFT: A multileader Byzantine fault tolerance protocol for high throughput," in *Proc. IEEE Int. Perf., Comput., Commun. Conf. (IPCCC)*, 2021, pp. 1–10. doi: 10.1109/IPCCC52140.2021.00007.

[4] D. Cason, E. Fynn, N. Milosevic, *et al.*, "The design, architecture and performance of the Tendermint blockchain network," in *Proc. 40th Int. Symp. Reliable Distrib. Syst. (SRDS)*, 2021, pp. 23–33. doi: 10.1109/SRDS53493.2021.00011.

[5] J. Karamachoski and L. Gavrilovska, "Tendermint performance with large transactions: The healthcare system scenario," in *Proc. Int. Balkan Conf. Commun. Netw. (BalkanCom)*, 2021, pp. 85–89. doi: 10.1109/BalkanCom51494.2021.9440667.

[6] S. K. Arora, G. Kumar, and T. Kim, "Blockchain based trust model using Tendermint in vehicular ad hoc networks," *Appl. Sci.*, vol. 11, no. 5, p. 1998, 2021. doi: 10.3390/app11051998.

[7] E. Buchman, "Tendermint: Byzantine fault tolerance in the age of blockchains," M.S. thesis, Univ. of Guelph, 2016.

[8] N. Lagaillardie, M. A. Djari, and Ö. Gürcan, "A computational study on fairness of the Tendermint blockchain protocol," *Information*, vol. 10, no. 12, p. 378, 2019. doi: 10.3390/info10120378.

[9] S. Braithwaite, E. Buchman, I. Khoffi, *et al.*, "A Tendermint light client," *arXiv* preprint arXiv:2010.07031, 2020. [Online]. Available: https://arxiv.org/abs/2010.07031

[10] Y. Amoussou-Guenou, A. Del Pozzo, M. Potop-Butucaru, *et al.*, "Dissecting Tendermint," *arXiv* preprint arXiv:1809.09858, 2018. [Online]. Available: https://arxiv.org/abs/1809.09858

[11] Y. Amoussou-Guenou, B. Biais, M. Potop-Butucaru, *et al.*, "Rational vs Byzantine players in consensus-based blockchains," in *Proc. Int. Conf. Autonomous Agents Multiagent Syst. (AAMAS)*, 2020, pp. 43–51. doi: 10.1145/3371445.3372211.

[12] L. Lei, C. Lan, and L. Lin, "Chained Tendermint: A parallel BFT consensus mechanism," in *Proc. Int. Conf. Hot Inf.-Centric Netw. (HotICN)*, 2020, pp. 25–33. doi: 10.1109/HotICN50710.2020.00012.

[13] I. Gerasimov and I. Chizhov, "Consensus algorithm of Tendermint platform and proof of lock change mechanism," *Int. J. Open Inf. Technol.*, vol. 7, no. 6, pp. 24–29, 2019.

[14] E. Buchman, J. Kwon, and Z. Milosevic, "The latest gossip on BFT consensus," *arXiv* preprint arXiv:1807.04938, 2018. [Online]. Available: https://arxiv.org/abs/1807.04938

[15] S. Gao, T. Yu, J. Zhu, *et al.*, "T-PBFT: An EigenTrust-based practical Byzantine fault tolerance consensus algorithm," *China Commun.*, vol. 16, no. 12, pp. 111–123, 2019. doi: 10.23919/JCC.2019.12.012.

[16] B. Choi, J. Sohn, D. J. Han, *et al.*, "Scalable network-coded PBFT consensus algorithm," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2019, pp. 857–861. doi: 10.1109/ISIT.2019.8849713.

[17] L. Lao, X. Dai, B. Xiao, *et al.*, "G-PBFT: A location-based and scalable consensus protocol for IoT-blockchain applications," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. (IPDPS)*, 2020, pp. 664–673. doi: 10.1109/IPDPS47981.2020.00097.

[18] Y. Wu, P. Song, and F. Wang, "Hybrid consensus algorithm optimization: A mathematical method based on POS and PBFT and its application in blockchain," *Math. Probl. Eng.*, vol. 2020, Art. no. 3459352, 2020. doi: 10.1155/2020/3459352.

[19] I. M. Coelho, V. N. Coelho, R. P. Araujo, *et al.*, "Challenges of PBFT-inspired consensus for blockchain and enhancements over Neo DBFT," *Future Internet*, vol. 12, no. 8, p. 129, 2020. doi: 10.3390/fi12080129.

[20] Y. Meshcheryakov, A. Melman, O. Evsutin, *et al.*, "On performance of PBFT blockchain consensus algorithm for IoT-applications with constrained devices," *IEEE Access*, vol. 9, pp. 80559–80570, 2021. doi: 10.1109/ACCESS.2021.3087113.

[21] X. Zheng, W. Feng, M. Huang, *et al.*, "Optimization of PBFT algorithm based on improved C4.5," *Math. Probl. Eng.*, vol. 2021, Art. no. 6480931, 2021. doi: 10.1155/2021/6480931.

[22] Y. Kim and J. Park, "Hybrid decentralized PBFT blockchain framework for OpenStack message queue," *Hum.-Centric Comput. Inf. Sci.*, vol. 10, no. 1, pp. 1–12, 2020. doi: 10.1186/s13673-020-00233-y.

[23] G. Xu, H. Bai, J. Xing, *et al.*, "SG-PBFT: A secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles," *J. Parallel Distrib. Comput.*, vol. 164, pp. 1–11, 2022. doi: 10.1016/j.jpdc.2022.01.008.

[24] J. Chen, X. Zhang, and P. Shangguan, "Improved PBFT algorithm based on reputation and voting mechanism," in *J. Phys.: Conf. Ser.*, vol. 1486, no. 3, p. 032023, 2020. doi: 10.1088/1742-6596/1486/3/032023.

[25] X. Feng, J. Ma, Y. Miao, *et al.*, "Social characteristic-based propagation-efficient PBFT protocol to broadcast in unstructured overlay networks," *IEEE Trans. Dependable Secure Comput.*, 2021. doi: 10.1109/TDSC.2021.3069052.

[26] S. Zhu, Z. Zhang, L. Chen, *et al.*, "A PBFT consensus scheme with reputation value voting based on dynamic clustering," in *Int. Conf. Security Privacy Digit. Econ.*, 2020, pp. 336–354. doi: 10.1007/978-3-030-46232-0_30

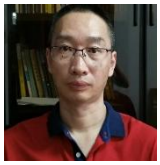
AUTHORS' PROFILE



Zhiyong Liang holds a master's degree and is a senior laboratory engineer. He currently serves as the Deputy Director of the Computer Center at University of Sanya. His main research interests include blockchain technology and network information security. He has undertaken two research projects, published eight academic papers, authored three textbooks, and applied for four software copyrights.



Rongwang Jiang holds a master's degree and is an associate professor with a background in computer science. He possesses both solid theoretical knowledge and extensive practical experience. He has led provincial-level natural science foundation projects and educational reform initiatives funded by the provincial education department, and has published multiple core journal papers, achieving notable research outcomes. At University of Sanya, he serves as the Deputy Director of the Computer Teaching Department, where he promotes teaching innovation and has guided students to win third prize in the national finals of the China Computer Design Competition, demonstrating strength in both teaching and research.



Ming Yang received the B.S. degree in computer software from Shandong University, China, the M.S. degree in computer applications from the Nanjing Institute of Communication Engineering, China, and the Ph.D. in communication and information systems from PLA University of Science and Technology, China, in 1989, 1995, and 2000, respectively. His research interest includes information security and privacy protection.



Boxiong Yang is a Professor at the University of Sanya, China. He received the B.S. degree in Computer Software from Huazhong Normal University, China, the M.S. degree in Geophysics from the Institute of Seismology, Seismological Bureau, China, and the Ph.D. in Geophysics from the Institute of Geophysics, China Seismological Bureau in 1997, 2000, and 2005, respectively. His area of research interest includes Photogrammetry, Computer Vision, and Artificial Intelligence.