

JOURNAL OF ENGINEERING RESEARCH & SCIENCES

JENRS



www.jenrs.com
ISSN: 2831-4085

Volume 5 Issue 6
June 2026

EDITORIAL BOARD

Editor-in-Chief(s)

Dr. Jinhua Xiao

Department of Industrial Management
Politecnico di Milano, Italy

Prof. Rashid A Saeed

Scientific Research Deanship, Lusail
University, Qatar

Prof. Abdeltif Amrane

Institute of Chemical Sciences of Rennes,
University of Rennes, France

Prof. Chi-Wai Chow

Department of Photonics, National Yang Ming
Chiao Tung University, Taiwan

Prof. Hui Liu

School of Artificial Intelligence, Nanjing
University of Information Science and
Technology, China

Editorial Board Members

Dr. Jianhang Shi

Department of Chemical and Biomolecular
Engineering, The Ohio State University, USA

Dr. Jianhui Li

Molecular Biophysics and Biochemistry,
Yale University, USA

Dr. Sonal Agrawal

Rush Alzheimer's Disease Center, Rush
University Medical Center, USA

Dr. Lixin Wang

Department of Computer Science,
Columbus State University, USA

Prof. Kamran Iqbal

Department of Systems Engineering, University
of Arkansas Little Rock, USA

Dr. Unnati Sunilkumar Shah

Department of Computer Science, Utica
University, USA

Dr. Anna Formica

National Research Council, Istituto di Analisi dei
Sistemi ed Informatica, Italy

Dr. Ramcharan Singh Angom

Biochemistry and Molecular Biology,
Mayo Clinic, USA

Prof. Anle Mu

School of Mechanical and Precision Instrument
Engineering, Xi'an University of Technology,
China

Dr. Prabhask Dadhich

Biomedical Research, CellBio, USA

Dr. Qichun Zhang

Department of Computer Science, University of
Bradford, UK

Dr. Qiong Chen

Navigation College, Jimei University, China

Dr. Żywiołek Justyna

Faculty of Management, Czestochowa University
of Technology, Poland

Dr. Mingsen Pan

University of Texas at Arlington, USA

Dr. Diego Cristallini

Department of Signal Processing & Imaging
Radar, Fraunhofer FHR, Germany

Ms. Madhuri Inupakutika

Department of Biological Science, University of
North Texas, USA

Dr. Deepak Bhaskar Acharya

Department of Computer Science, The University
of Alabama in Huntsville, USA

Dr. Gabriel-Alexandru Constantin

Department of Biotechnical Systems, Faculty of
Biotechnical Systems Engineering, National
University of Science and Technology
POLITEHNICA Bucharest, Romania

Prof. Cheng-Chi Lee

Department of Library and Information Science,
Fu Jen Catholic University, Taiwan

Prof. Marian Pompiliu Cristescu

Finance Accounting Department, Lucian Blaga
University of Sibiu, Romania

Dr. Shabir Ahmad

Department of Mathematics and Physics,
University of Campania Luigi Vanvitelli, Italy

Dr. Serdar Halis

Department of Automotive Engineering,
Pamukkale University, Turkey

Dr. Sarat Chandra Mohapatra

Centre for Marine Technology and Ocean
Engineering (CENTEC), Instituto Superior
Técnico/University of Lisbon, Portugal

Dr. Amin Amiri Delouei

Department of Mechanical Engineering,
University of Bojnord, Iran

Dr. Alexander Chupin

Faculty of Economics, RUDN University, Russia

Dr. Haiping Xu

Computer and Information Science
Department, University of Massachusetts
Dartmouth, USA

Prof. Hamid Mattiello

Department of Business and Economics,
University of Applied Sciences (FHM),
Germany

Dr. Ali Golestani Shishvan

Department of Electrical & Computer
Engineering, University of Toronto,
Canada

Prof. Ahmad M. A. zamil

Department of Marketing, Prince Sattam
bin Abdulaziz University, Saudi Arabia

Dr. Lilik Jamilatul Awal

Faculty of Advanced Technology and
Multidiscipline, Airlangga University,
Indonesia

Dr. Behrokh Beiranvand

TEKsystems at Apple Inc, Contractor at
Apple Inc, United States

Prof. Giuseppe Oliveto

Department of Engineering, University of
Basilicata, Italy

Dr. Saad khadar

Electrical Engineering Department,
University of Djelfa, Algeria

Dr. Ali Moghassemi

Electrical Engineering, University of
Wisconsin-Milwaukee, United States

Dr. Fan Xu

Shenzhen Institute for Advanced Study,
University of Electronic Science and
Technology of China, China

Prof. Juan Eduardo Nápoles Valdes

Matemáticas, Universidad Nacional del
Nordeste, Argentina

Prof. Wafaa Mohammed Ridha

Technical Institute of Babylon, Al-Furat Al-Awsat
Technical University, Iraq

Dr. Parveen Berwal

Civil Engineering, Galgotias College of
Engineering and Technology, Greater
Noida, India

Prof. Filipe Almeida Correa do Nascimento

Transportation Engineering Program, Instituto
Militar de Engenharia (Military Institute of
Engineering), Brazil

Dr. Marius Stef

Department of Physics, West University of
Timisoara, Romania

Mr. Anderson Apolônio Lira Queiroz

Center Computer, Universidade Federal Pernambuco,
Brazil

Dr. George Dănuț Mocanu

Team Sports Games and Physical
Education, Dunărea de Jos University,
Romania

Dr. Sachin Kumar

Electronics and Communication Engineering,
Galgotias College of Engineering and Technology,
India

Dr. André Saandim

Departamento de Ciências Florestais,
Universidade de Trás-os-Montes e Alto
Douro, Portugal

Dr. Ram Prasad

Department of Botany, Mahatma Gandhi Central
University, India

Prof. Juan Antonio López Ramos

Department of Mathematics, University of
Almeria, Spain

Dr. Juan Molina

Departamento de Biología Bioquímica y
Farmacia, Universidad Nacional del Sur,
Argentina

Dr. Hanan Mikhael Dawood Habbi

Department of Electrical Engineering,
University of Baghdad, Iraq

Prof. Alexander E. Hramov

Research Institute of Applied AI and Digital
Solutions, Plekhanov Russian University of
Economics, Russia

Prof. Aissani Amar

Dept Artificial Intelligence & Data Science,
University of Science & Technology Houari
Boumediene (USTHB), Algeria

Dr. Alina Alb Lupas

Department of Mathematics and Computer
Science, University of Oradea, Romania

Prof. Rabha W. Ibrahim

Develop Researchs Department, SAS,
United States

Prof. Waluyo

Department of Electrical Engineering, Institut
Teknologi Nasional Bandung, Indonesia

Dr. Fathurrahman Lananan

Faculty of Bioresources and Food
Industries, Universiti Sultan Zainal Abidin
(UniSZA), Malaysia

Prof. Marco Milanese

Department of Engineering for Innovation,
University of Salento, Italy

Dr. Bhupendra Kumar Singh

Division of Advanced Nuclear Engineering,
Pohang University of Science and
Technology (POSTECH), South Korea

Dr. Seyit Uguz

Department of Biosystems Engineering, Yozgat
Bozok University, Turkey

Dr. Alejandro Medina Santiago

Computer Science, Institute National of
Astrophysic, Optics and Electronics, Mexico

Prof. Rupesh Kumar

Jindal Global Business School, O P Jindal Global
University, India

Prof. Laura Eugenia Paulette

Faculty of Agriculture, Technical and soil
sciences, University of Agricultural Sciences and
Veterinary Medicine Cluj Napoca, Romania

Dr. Ana Maria Mihaela Iordache

Informatics, Statistics and Mathematics,
Romanian American University, Romania

Dr. V.I. Zhukov

Department of Chemistry and Chemical
Technology, Novosibirsk State Technical
University, Russia

Dr. Ammar Mohammad Jamil Odeh

King Hussein School of Computing Sciences,
Princess Sumaya University for Technology,
Jordan

Prof. Pshtiwan Othman Mohammed

College of Education, University of Sulaimani,
Iraq

Dr. Alex Rizzato

Department of Biomedical Sciences, University of
Padova, Italy

Dr. Farrukh Shahzad

School of Economics and Management,
Guangdong University of Petrochemical
Technology, China

Dr. Esra Calik Bayazit

Computer Engineering, Fatih Sultan Mehmet
Vakif University

Dr. Fazlur Rahman

Faculty of Built Environment and
Surveying, Universiti Teknologi Malaysia,
Malaysia

Dr. Laura Gioiella

School of Architecture and Design,
University of Camerino, Italy

Dr. Hamzeh Mehrabi

College of Science, University of Tehran,
Iran

Dr. Hakim Mellah

Computer Science and Software
Engineering Department, Concordia
University, Canada

Dr. Maha AbouBakr Ibrahim

Faculty of Engineering, Architectural
engineering department, Misr University
for Science and Technology, Egypt

Prof. Maged S. Al-Fakeh

Department of Chemistry, Qassim
University, Saudi Arabia

Prof. Boris F. Minaev

Arrenius Laboratory, Uppsala University,
Sweden

Dr. Ermelinda Kordha

Department of Marketing and Tourism,
University of Tirana, Albania

Prof. Francesco Inchingolo

Interdisciplinary of Medicine, University of
Bari Aldo Moro, Italy

Prof. Alban Kuriqi

Civil Engineering, University for Business
and Technology

Dr. Papa Pio Ascona Garcia

Profesional De Ingenieria Civil,
Universidad Nacional Intercultural,
Fabiola Salazar Leguía

Prof. Osamah Ibrahim Khalaf

Al-Nahrain Renewable Energy Research Center,
Al-Nahrain University

Prof. Acácio Manuel Raposo Amaral

Coimbra Institute of Engineering, Polytechnic
Institute of Coimbra

Dr. A B M Amrul Kaish

Department of Civil Engineering, Universiti
Kebangsaan Malaysia

Prof. Vitalii Ivanov

Manufacturing Engineering, Machines and Tools,
Sumy State University

Ms. Beknazarova Saida Safibullayevna

Television and media technologies, Tashkent
university of information technologies

Prof. Chow Ming Fai

Department of Civil Engineering, Monash
University Malaysia

Dr. Mojtaba Fardi

Department of Applied Mathematics, Shahrekord
University

Dr. Takele Ferede Agajie

School of Electrical and Computer Engineering,
Institute of Technology, Debre Markos University

Prof. Ebenezer Esenogho

Centre for Artificial Intelligence and
Multidisciplinary Innovations, University of
South Africa

Dr. Lim Chen Kim

Institute of Environment and Development
(LESTARI), Universiti Kebangsaan Malaysia
(UKM)

Mr. Sushant K. Rawal

Department of Mechanical Engineering,
McMaster University

Prof. Wael A. Altabey

Department of Mechanical Engineering,
Alexandria University

Dr. Adeb Ali Mohammed Ahmed Al-Samet

Faculty of Information and
Communication Technology, Universiti
Tunku Abdul Rahman

Dr. Abhishek Phadke

School of Engineering and Computing,
Christopher Newport University

Dr. Zishan Shaikh

Department of Environmental Science,
Savitribai Phule Pune University

Dr. Sofiane HADDAD

Electronic Department, MSB Jijel
University

Dr. Tibor Krenicky

Faculty of Manufacturing Technologies
with a seat in Prešov, Technical University
of Košice

Prof. Denys Baranovskyi

Department of Computerization and
Robotization of Industrial Processes,
Rzeszow University of Technology

Prof. Mejdi Snoussi

College of Science, Biology Department,
University of Hail

Prof. YoungPak Lee

Physics/Optical Science and Engineering,
Hanyang University

Prof. Mohamed Mohamed Zaky Ahmed

College of Engineering/Department of
Mechanical Engineering, Prince Sattam
bin Abulaziz University

Dr. Maryna Bulakh

Department of Computerization and
Robotization of Industrial Processes,
Rzeszow University of Technology

Dr. Crescenzo Pepe

Dipartimento di Ingegneria dell'Informazione,
Università Politecnica delle Marche

Prof. Bucur Daniel

Department: Pedotechnics, Iasi University of Life
Sciences

Dr. Muhammad Imran Khan

Institute of Soil and Environmental Sciences,
University of Agriculture Faisalabad

Mr. Noor Mohammad Mohammad

School of Mechanical Engineering, Purdue
University

Dr. Habib Ur Rahman Habib

University of Aberdeen

Dr. Philip Adewuyi

Department of Electrical Engineering, College of
Engineering, Wigwe University

Dr. Hadi Monsef Ahmadi

Department of Engineering, Niccolò Cusano
University

Prof. Ion Sandu

Arheoinvest Research Platform, Alexandru
Ioan Cuza University of Iasi

Dr. Naeem Saleem

Department of Mathematics, University of
Management and Technology

Dr. Israr Ahmad

Coppin State University

Dr. Pramod Gawal

Department of Chemistry, College of
Natural Sciences, Pusan National
University

Ms. Sandhya Kumari

Department of Pharmacy, Lucknow Model
College of Pharmacy

Dr. Yulia Kumar

Department of Computer Science and
Technology, College of Business and
Public Management, Kean University

Editorial

The Journal of Engineering Research and Sciences (JENRS) continues to provide a multidisciplinary platform for research addressing emerging challenges across engineering, science, and technology. The contributions presented in this issue reflect the increasingly interconnected nature of contemporary engineering research, spanning cybersecurity, human-computer interaction, condensed matter physics, artificial intelligence, public-sector digital governance, and intelligent fault diagnosis. Collectively, these studies demonstrate how theoretical developments, systematic reviews, experimental validation, and emerging computational methodologies can contribute to the development of more secure, reliable, accessible, and intelligent technological systems.

The security of legacy network infrastructure remains a critical concern, particularly within highly regulated financial and mainframe environments. A systematic migration framework from SNMPv2c to SNMPv3 is presented for IBM z/OS, Cisco IOS-XE, SolarWinds, and related operating-system environments, incorporating authenticated privacy mechanisms based on HMAC-SHA-96 and AES-128. Through FAIR-based risk analysis, the proposed transition demonstrates a substantial reduction in expected annual loss, while the five-phase migration methodology addresses coexistence, asset discovery, RACF configuration, engineID management, and cutover validation. The alignment of the approach with established security standards and guidance further enhances its practical value for organizations seeking to modernize legacy network-management infrastructures [1].

Accessibility and engagement in user experience evaluation are examined through the development of voice-based adaptations of the User Experience Questionnaire. Two conversational implementations, Voice1-7 and Voice2Q, together with the visually supported Voice2Q+VC variant, demonstrate how natural language interaction can be incorporated into established UX assessment methodologies. Validation involving 345 participants indicates that the voice-based approaches can produce results comparable to the conventional written questionnaire, while Voice2Q improves response consistency and Voice2Q+VC helps address cognitive workload associated with voice-only interaction. These findings highlight the potential of conversational interfaces to make UX evaluation more accessible while maintaining measurement validity and improving user engagement [2].

The fundamental mechanisms governing superconducting critical temperature remain an important challenge in condensed matter physics. A phenomenological framework based on the concepts of superconducting gene, superconducting DNA, orbital electron-hole pairs, and associated structural parameters is proposed to describe variations in critical temperature across copper oxides, iron-based superconductors, and high-pressure hydrides. The resulting orbital engineering strategy considers electron orbital filling, light-element heterojunctions, pressure, doping, and isotope effects as potential pathways for increasing superconducting transition temperatures. By seeking a unified description across different superconducting material families, the study offers theoretical perspectives that may support the future design of high-temperature superconducting materials and technologies [3].

The growing availability of social media content in Bengali presents both an opportunity and a challenge for computational emotion analysis, particularly because of limited annotated resources and the informal nature of online language. A unified six-class Bengali emotion classification framework based on a manually annotated corpus of 5,401 social media comments systematically evaluates recurrent, transformer-based, and hybrid architectures. The proposed soft voting ensemble combines complementary contextual representations and achieves 91.31% accuracy with a weighted F1 score of 0.913, outperforming individual models. The incorporation

of LIME and SHAP further strengthens interpretability, contributing to the development of more transparent and reliable natural language processing systems for Bengali-language applications [4].

Digital transformation within public-sector organizations requires technological innovation to be balanced with governance, regulatory, and operational requirements. A systematic review of 39 sources examines the potential integration of Artificial Intelligence for IT Operations and Machine Learning Operations into ITIL-based change and release management workflows. The analysis identifies key themes involving AI-driven monitoring and incident management, public-sector governance constraints, legacy-system challenges, and limited adoption of AI within change and release processes. The identified gaps in explainability, empirical validation, AI-ready change records, and governance frameworks provide a valuable foundation for future research aimed at developing responsible and effective AI-supported operational management in public-sector environments [5].

The increasing complexity of cyber-physical systems has created a growing need for intelligent fault-diagnosis methods capable of understanding relationships among interconnected components. A comprehensive survey of Graph Neural Network-based fault diagnosis examines graph representations, learning paradigms, diagnostic objectives, deployment architectures, benchmark datasets, and evaluation practices across industrial automation, smart grids, transportation, healthcare, and other domains. Particular attention is given to practical challenges including topology drift, cascading misdiagnosis, noise amplification, open-set misclassification, adversarial vulnerability, and concept drift. The discussion of physics-informed learning, multimodal fusion, dynamic graph modeling, and privacy-preserving federated approaches, together with practitioner-oriented design guidance, provides a broad perspective on the path toward reliable and deployable GNN-based diagnostic systems for safety-critical infrastructures [6].

The contributions assembled in this issue demonstrate the broad scope and evolving character of engineering research, where cybersecurity, artificial intelligence, digital governance, human-centered technologies, advanced materials, and intelligent diagnostics increasingly intersect. From securing legacy infrastructure and improving digital accessibility to exploring new theoretical pathways for superconductivity and developing robust AI-driven analytical systems, the featured studies address challenges of both scientific significance and practical relevance. The collective findings provide valuable insights for researchers, practitioners, and technology developers and underscore the importance of interdisciplinary approaches in advancing reliable, secure, sustainable, and intelligent engineering systems.

References:

- [1] R.K. Shaw, "SNMPv2-to-SNMPv3 Migration in Financial Mainframes: Risk Reduction and Repeatable Playbook," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 1–14, 2026, doi:10.55708/js0506001.
- [2] I. Diaz-Oreiro, G. Lopez, "Beyond Written Surveys: Validating Voice-Based Implementations of the User Experience Questionnaire," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 15–26, 2026, doi:10.55708/js0506002.
- [3] G. Zhang, X. Zhang, "Superconductivity Research Method Based on Atomic Properties and Hole-Electron Tensor," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 27–38, 2026, doi:10.55708/js0506003.
- [4] M. Alam, S.S. Aonty, S.N. Mahmud, N.R. Swachha, A.T. Wazih, "Bengali Emotion Classification from Social Media Text Using Deep Learning and Transformer-Based Models with Explainability," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 39–55, 2026, doi:10.55708/js0506004.

- [5] C. Ganduri, "From ITIL to AIOps in Public Sector: A Systematic Literature Review," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 56–66, 2026, doi:10.55708/js0506005.
- [6] V. Tiwari, O. Suaifan, R. Othman, A. Gupta, "Graph Neural Networks for Fault Diagnostics in Cyber-Physical Systems: A Survey of Taxonomy, Deployment Architectures and Failure Modes," *Journal of Engineering Research and Sciences*, vol. 5, no. 6, pp. 67–96, 2026, doi:10.55708/js0506006.

Editor-in-chief

Dr. Jinhua Xiao

CONTENTS

<i>SNMPv2-to-SNMPv3 Migration in Financial Mainframes: Risk Reduction and Repeatable Playbook</i> Rohit Kumar Shaw	01
<i>Beyond Written Surveys: Validating Voice-Based Implementations of the User Experience Questionnaire</i> Ignacio Diaz-Oreiro and Gustavo Lopez	15
<i>Superconductivity Research Method Based on Atomic Properties and Hole-Electron Tensor</i> Guohua Zhang and Xiyu Zhang	27
<i>Bengali Emotion Classification from Social Media Text Using Deep Learning and Transformer-Based Models with Explainability</i> Mujtahid Alam, Shuhena Salam Aonty, Sha Newaz Mahmud, Nahid Riaz Swachha and Ahmed Talal Wazih	39
<i>From ITIL to AIOps in Public Sector: A Systematic Literature Review</i> Catherine Ganduri	56
<i>Graph Neural Networks for Fault Diagnostics in Cyber-Physical Systems: A Survey of Taxonomy, Deployment Architectures and Failure Modes</i> Vaibhavi Tiwari, Ola Suaifan, Ramy Othman and Anand Gupta	67

SNMPv2-to-SNMPv3 Migration in Financial Mainframes: Risk Reduction and Repeatable Playbook

Rohit Kumar Shaw* 

Infrastructure Engineering, Wylie, TX, 75098, United States of America

*Corresponding Author: Rohit Kumar Shaw, rohitshaw.infosec@gmail.com

ABSTRACT: Legacy SNMP versions 1 and 2c send community strings in cleartext with no authentication or encryption. In financial mainframe environments that operate IBM z/OS and are tied to Cisco IOS equipment, attackers actively exploit SNMPv2c as an attack vector. This paper quantifies the reduction in risk that can be achieved by deploying SNMPv3 in IBM z/OS 2.5, Cisco IOS-XE, SolarWinds network management systems, and a variety of operating-system environments, using the authenticated privacy (authPriv) mode defined in RFCs 3411 to 3415 and 3826, as well as HMAC-SHA-96 and AES-128. Each of these platforms outlines and verifies a five-phase, repeatable migration process. Using the FAIR (Factor Analysis of Information Risk) method of risk analysis, the loss from the exploitation of SNMP in mainframe environments indicates a 90 to 96 percent reduction in expected annual loss under stated assumptions (a drop from \$1.1M to \$4.5M under SNMPv2c to \$40K to \$250K in SNMPv3). This methodology covers these important points in z/OS migration: coexistence with an existing system, asset discovery and documentation of your RACF settings and views, design of your engineID management plan, and validation of your cutover strategy. This paper aligns with the PCI-DSS v 4.0, NIST SP 800-53 Rev. 5, and NSA/CISA guidance for mainframe migrations and includes a checklist that security architects and mainframe operations teams can apply while implementing and validating the migration process.

KEYWORDS: SNMP, SNMPv3, SNMPv2c, IBM z/OS, User-based Security Model (USM), View-based Access Control Model (VACM), network security, mainframe, Resource Access Control Facility (RACF), authPriv, AES-128, risk reduction, PCI-DSS, NIST SP 800-53.

1. Introduction

The Simple Network Management Protocol, or SNMP, is a fundamental management protocol used for monitoring network systems in large enterprises. At banks and other financial services organizations that rely on a core transaction process based on the IBM z/OS mainframe operating system, SNMP management agents are running on the z/OS Communications Server itself, on Cisco routers and switches at the network periphery, and on a range of other equipment such as load balancing devices and storage subsystems. Network monitoring products from Omegamon and SunGard/IBM products like NetView monitor SNMP-enabled z/OS systems and third-party equipment, such as Cisco devices, SolarWinds, and Zabbix, while polling usage and processor loads on network interfaces, TCP connection counts, and system availability [1].

Despite this critical role, the majority of deployed SNMP configurations in legacy financial environments continue to operate version 1 or version 2c. Both versions authenticate solely through a shared community string transmitted in plaintext over UDP port 161. Any observer with access to the network path between the management station and the managed device can read this string using standard packet capture tools [2]. With a valid community string, an attacker can enumerate MIBs to gather detailed device topology information and, with read-write access, can alter device configuration [3]. These are not theoretical threats. The Russian GRU-affiliated threat actor APT28 exploited default SNMPv2c community strings on Cisco routers belonging to U.S. government agencies and Ukrainian organizations from 2021 through 2023, injecting persistent malware and exfiltrating routing tables using nothing more than a default community string of "public."

Financial mainframe environments present an especially high risk profile relative to general-purpose IT. The systems process significant amounts of cardholder and ACH transaction data continuously, with strict change windows (24x7) limiting when urgent patches may be applied and sit at the intersection of PCI-DSS and SOX compliance obligations. Therefore, an SNMP-based breach of z/OS infrastructure will expose network topology as well as the audit trail, transaction processing engine, and compliance attestation record simultaneously.

SNMPv3, standardized in December 2002 as IETF Internet Standard STD 62 through RFCs 3411 to 3418 [4], [5], [6], [1], addresses every principal security threat that undermines SNMPv2c. Its User-based Security Model (USM) provides per-user HMAC authentication and symmetric-key encryption. Its View-based Access Control Model (VACM) enforces least-privilege MIB access. The timeliness module prevents replay attacks by enforcing a 150-second clock synchronization window.

In summary, this paper makes the following contributions. We provide a side-by-side security model comparison (both SNMPv2c and SNMPv3) at a protocol level based on RFCs. We analyze the current security risk post-migration based on a government's vulnerability advisory and CVE databases. We quantify the risk after migration using the Factor Analysis of Information Risk (FAIR) model with assumptions consistent with a typical financial mainframe environment. We provide detailed migration procedures, including the design pattern for the View-based Access Control Model (VACM), as per specific major equipment vendors' implementation instructions: IBM z/OS and Cisco IOS. Finally, we show how these procedures comply with requirements set by PCI DSS (v4.0), NIST SP 800-53 (Rev 5), and NSA CISA. We finish off the paper with a security post-migration checklist.

2. SNMPv2c vs. SNMPv3: Protocol Security Comparison

2.1. A. SNMPv2c Community-Based Security Model

The SNMPv2 community string model includes additional protocol-defined functionality for the community model, such as GetBulk and InformRequest, compared to SNMPv1 as defined in IETF RFC1901 [1], but it does not improve the security architecture of SNMPv1. The community string serves as the combination of a group ID, a password, and an access level (R/O) selection function. The community's name in SNMPv2c is implemented as an OCTET STRING without any application-level hashing, symmetric encryption, or message authentication mechanism in the SNMP protocol header. SNMP community strings are assumed to be static and not rotated by users or administrators; there are defaults ("public" read-only, "private" read/write) set

into devices that allow default community names with default access values. This is a massive surface targeted by previous APT28 cyber-attacks.

SNMPv2c lacks protection against any of these five key threats: Threat to Data (i.e., the integrity of the data in SNMP messages during transit), Threat to Message Source Identity (i.e., masquerade), Threat to Message Confidentiality (i.e., disclosure), Threat to Data Stream (i.e., the deletion and replay of messages), Denial-of-Service threat.

The primary vulnerabilities associated with both SNMPv1 and SNMPv2c can be easily summarized: the community string is sent in clear text via UDP, thus enabling anyone who monitors this traffic to capture it using common packet capture software; there is no verification of each message that is received by an administration workstation, therefore all messages could have been generated from anywhere; the content of all MIBs polled and the configuration of devices are in plain sight during transit because they are not encrypted; instead of being assigned to individual users, all access credentials for a network's management workstations are shared among all management workstations; and the two most commonly used community strings (the public string and the private string), which were set at the time of manufacture and are very rarely modified, create an always-available and well-known attack opportunity. All of these issues have also demonstrated practical vulnerability through CVE-2017-6742 [7] and the APT28 Jaguar Tooth campaign [8].

2.2. B. SNMPv3 User-Based Security Model

SNMPv3 fixes all the shortcomings of SNMPv2c by making three changes to its security. First, it provides authentication via an HMAC-SHA-96-based hash using the community string as a user-specific, engine-specific key; therefore, each message received is tied to a particular user ID and can't be spoofed without access to that user's local key. Second, AES-128 encryption in Cipher Feedback (CFB) mode protects the scoped PDU from being read on receipt, thereby rendering any capture of the encapsulated UDP traffic useful for nothing other than ciphertext, which includes no discernible community string or MIB information. Third, all messages are integrity checked by including a 96-bit truncated HMAC in the authenticated portion of every message received. This allows any modification made while the message was in transit to be identified before the message is processed.

In RFC 3411 [4], SNMPv3 was defined to use a modular architecture consisting of an SNMPv3 Application, Dispatcher Message Processing Subsystem, SNMPv3 Security Subsystem, and SNMPv3 Access Control Subsystem. An implementation of the SNMPv3 Security Subsystem is described in USM. Each user is identified by

a (userName, snmpEngineID) tuple and possesses distinct authentication credentials and, optionally, privacy credentials. Three security levels are defined: noAuthNoPriv (no protection, used for discovery only), authNoPriv (authentication without encryption), and authPriv (authentication plus encryption). Financial mainframe environments should operate exclusively in authPriv mode.

Authentication uses HMAC-MD5-96 or HMAC-SHA-96 [5]. The SHA variant computes an HMAC-SHA-1 over the entire serialized message using a 160-bit localized key, then truncates the result to 96 bits for insertion into the msg Authentication Parameters field. The localized key is derived per RFC 3414 section 2.6: the user password is expanded to fill 1,048,576 bytes (1 MB) by cyclic repetition, hashed to produce a master key Ku, and then hashed again with the snmp Engine ID concatenated on both sides to produce the localized key Kul. This means that compromise of credentials at one engine does not expose credentials used at any other engine.

Privacy uses CBC-DES (RFC 3414 [5], included for backward compatibility but cryptographically deprecated) or AES-CFB-128 (RFC 3826 [9]). AES-128 in CFB mode constructs its initialization vector from the 32-bit snmp Engine Boots value, the 32-bit snmp EngineTime value, and a 64-bit local salt. The salt is incremented with each message, ensuring IV uniqueness even when two messages are sent within the same second. AES-128 should be the minimum privacy protocol deployed; DES must be disabled. The replay attack window is enforced by requiring that the absolute difference between the received msgAuthoritativeEngineTime and the local notion of that engine time not exceed 150 seconds (RFC 3414 section 2.2.3). Past research has established that even fully configured SNMPv3 systems still have residual attack surfaces, such as the discovery mechanism for negotiating SNMP engine IDs and key localization values [10].

2.3. Feature Comparison Table

Table 1 summarizes the security feature comparison between SNMPv2c and SNMPv3.

Table 1: SNMPv2c vs. SNMPv3 Security Feature Comparison

Feature	SNMPv2c	SNMPv3 (authPriv)	RFC Ref.
Authentication Model	Community string (plaintext)	Per-user HMAC-SHA-96 or HMAC-MD5-96	RFC 3414
Encryption	None	AES-128 CFB or CBC-DES	RFC 3826
Integrity Protection	None	HMAC truncated to 96 bits per message	RFC 3414

Replay Protection	None	150-second timeliness window; engine boots counter	RFC 3414
Access Control	Community string maps to read/write	VACM: group, context, security level, MIB view	RFC 3415
Credential Scope	Shared across all management stations	Per-user, per-engine; localized keys	RFC 3414
MIB Visibility	Full or community-filtered	Fine-grained OID subtree views via VACM	RFC 3415
Notification Security	None	Authenticated Inform PDUs with retry	RFC 3413
Default Credentials Risk	High: "public"/"private" widely deployed	None: no default users or keys defined	RFC 3411

3. SNMPv3 Architecture and Security Models

3.1. Message Processing Architecture

Figure 1 illustrates the layered security architecture of an SNMPv3 message as specified in RFCs 3411 and 3414 [4], [5]. The outer message envelope carries the protocol version, message identifier, maximum message size, security flags (authFlag, privFlag, and reportableFlag packed into a single octet), the security model identifier (3 for USM), and the serialized security parameters. When a PDU goes through the scope, the scope ID is authenticated first, and the data is then encrypted to ensure message confidentiality and integrity.

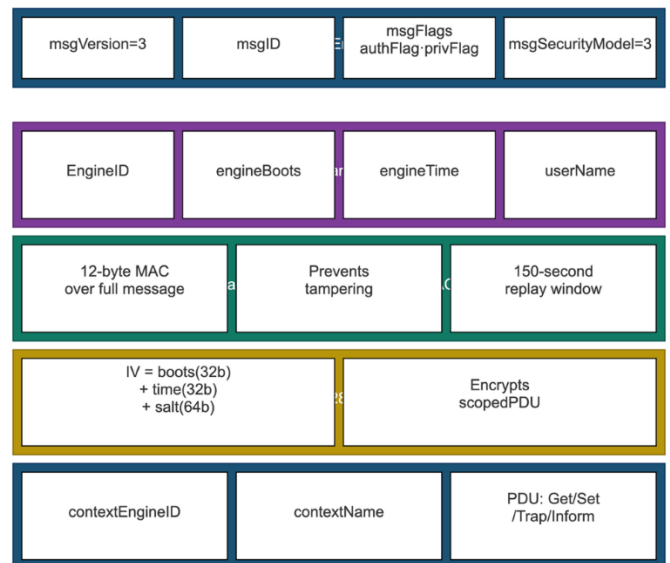


Figure 1: SNMPv3 USM Security Architecture and Message Processing (RFC 3414)

3.2. View-Based Access Control Model

VACM (RFC 3415) [6] implements a five-step access decision process operating on five MIB tables. The `vacmSecurityToGroupTable` stores a mapping from a role and security context to the name of a database group. The `AccessTable` contains a mapping from the following attributes of groups to the names of system views on the table: group name, context prefix, security model, required security level, `readView`, `writeView`, and `notifyView`. The `vacmViewTreeFamilyTable` specifies OID subtrees that each view includes or excludes, allowing administrators to restrict a monitoring user to system MIB and interface MIB OIDs while blocking access to security-sensitive subtrees. Figure 2 illustrates the VACM decision flow.

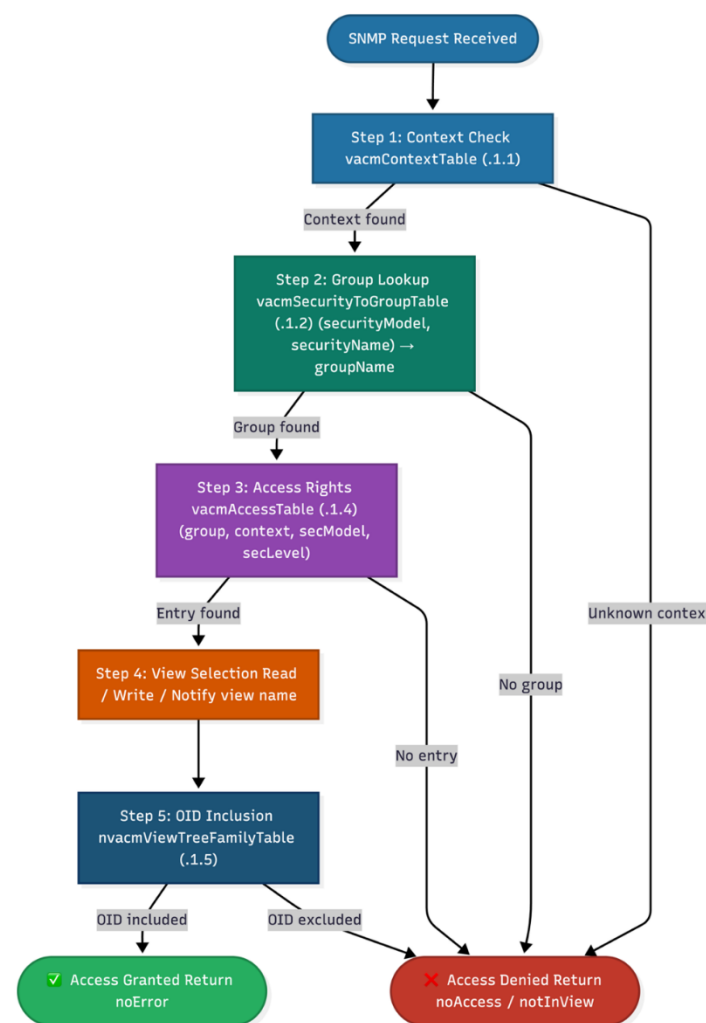


Figure 2: VACM Decision Flow for SNMPv3 Access Control (RFC 3415)

4. Risk Landscape: Pre-Migration Threat Analysis

4.1. Government and Industry Advisories

The U.S. Cybersecurity and Infrastructure Security Agency (CISA) published Alert TA17-156A in June 2017 [2], stating that "SNMPv3 should be the only version of SNMP employed" because SNMPv1 and SNMPv2c allow an adversary to sniff network traffic and recover community strings, enabling man-in-the-middle and

replay attacks. The NSA Network Infrastructure Security Guide [3] (October 2023) states that the NSA "highly discourages using SNMP version 1 or 2c" and dedicates Section 7.11.3 to requiring SNMPv3 for all remote device administration. CISA updated this position in December 2024, mandating that organizations "disable any unnecessary, unused, exploitable, or plaintext services and protocols, such as SNMPv1/v2c" [11] in the Enhanced Visibility and Hardening Guidance for Communications Infrastructure.

The most significant recent incident is documented in the April 2023 Joint Advisory issued by NCSC-UK, NSA, CISA, and the FBI [8]. APT28, a Russian GRU threat actor, exploited CVE-2017-6742 [7] against Cisco IOS routers from 2021 onward. The advisory states that affected routers were configured with SNMPv2c using weak or default community strings, including the default value "public." In at least 250 cases, including those targeting government entities in the United States, the strings were part of an attack called "Jaguar Tooth," in which APT28 injected malicious code into routers' memory, enabling the theft of routing table information and maintaining persistence after a router rebooted.

4.2. CVE Vulnerability Catalog

Table 2 presents the principal SNMP vulnerability catalog from 2002 through 2025, illustrating the persistent and evolving attack surface.

It is important to note the CVE-2017-6742 [7] and CVE-2025-20352 [12] affect the SNMP processing engine (not just community string parsing), so they exist for both SNMPv2c and SNMPv3 users (they are engine processing defects). It cannot be said strongly enough that any device that handles SNMP traffic must be quickly patched for these issues. The benefits of using SNMPv3 (and associated ACLs) should not be overlooked, though, since proper configuration makes network traffic unreachable except by the trusted network management systems' IP address, effectively closing that "back door."

4.3. Attack Taxonomy

Six primary attack categories target SNMPv1/v2c deployments in financial mainframe environments, which is shown in Table 3.

5. Network Architecture: Before and After Migration

Figure 3 contrasts the pre-migration and post-migration network management architectures.

In pre-migration architecture, UDP port 161 traffic between the NMS and any managed device carries the community string in the 7th field of the SNMPv2c message header. Any device on the Layer 2 segment, including other managed devices, can capture this traffic. In the post-migration architecture, the scoped PDU is AES-128

Table 2: Selected SNMP Vulnerability History (2002-2025)

CVE	Year	CVSS	Description	Affected	Exploited
CVE-2002-0013	2002	10.0	DoS/privilege escalation via SNMPv1 trap handling (PROTOS suite)	Multi-vendor	CERT CA-2002-03
CVE-2002-0012	2002	10.0	DoS/privilege escalation via SNMPv1 request handling	Multi-vendor	CERT CA-2002-03
CVE-2012-3268	2012	8.5	SNMP credential disclosure via CWE-522	HP/H3C/Huawei	Public PoC
CVE-2017-6742	2017	8.8	RCE via buffer overflow in SNMP MIBs on Cisco IOS/IOS XE	Cisco IOS/XE	APT28; CISA KEV
CVE-2020-15862	2020	8.8	Arbitrary command execution as root via EXTEND MIB in Net-SNMP	Net-SNMP <= 5.8	Active exploitation
CVE-2025-20352	2025	7.7	Stack-based buffer overflow in Cisco IOS/IOS XE SNMP subsystem	Up to 2M devices	CISA KEV Oct 2025
CVE-2025-68615	2025	9.8	Buffer overflow in Net-SNMP snmptrapd; no authentication required	Net-SNMP < 5.9.5	Critical: no auth needed

Table 3: Comparative analysis of SNMPv2c vulnerabilities and SNMPv3 mitigation strategies for network management security.

Attack Category	SNMPv2c Exposure	SNMPv3 Mitigation	Residual Risk
Community string sniff	Critical: plaintext UDP	Community strings eliminated; USM keys never transmitted	No Access Control List (ACLs) applied.
MIB enumeration	Full MIB tree exposed	VACM view restricts OID access per user role	Low-view design
Configuration tampering	Write access via RW community	writeView empty unless explicitly granted	Low with least privilege
Replay attack	Any captured PDU replayable	150-second timeliness window + boots counter	Negligible [10]
Engine buffer overflow	Exploitable by any IP with port 161 access	ACL restricts access to NMS IPs only	Medium: patch required
DDoS amplification	Public-facing UDP 161 exploitable	ACL blocks external access and information replace traps	Low with ACLs

encrypted. Even if the UDP traffic is captured, the attacker obtains only ciphertext, and the absence of community strings means there is no static secret to brute-force from offline capture.

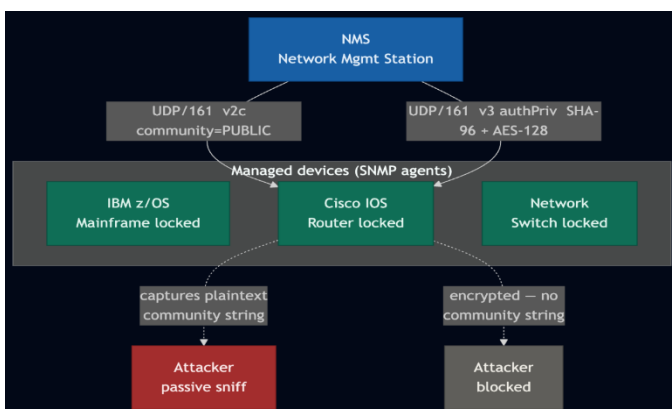


Figure 3: The SNMP architecture before and after migration. The left path shows SNMPv2c sending community strings in plain text, which makes it easy for someone to passively intercept them. The right path shows SNMPv3 authPriv with SHA-96 and AES-128, which gets rid of community strings and stops the attacker.

6. Risk Reduction Analysis

6.1. Risk Model and Assumptions

The analysis uses the FAIR (Factor Analysis of Information Risk) model [13], which decomposes risk into

Loss Event Frequency (LEF) and Loss Magnitude (LM). LEF is the product of Threat Event Frequency (TEF) and Vulnerability (V). LM spans six loss categories: productivity loss, response costs, equipment replacement, competitive advantage loss, fines and judgments, and reputation damage. The numbers provided reflect the cost of protecting a mid-sized financial institution (with 150 to 300 SNMP-enabled mainframe and network devices) that holds cardholder data and is required to comply with the Payment Card Industry Data Security Standards (PCI DSS) and SOX Section 404, as outlined in Table 4.

The parameters outlined above are based on a large financial organization with 150-300 SNMP-enabled devices and a \$50M mainframe that is within PCI DSS scope. The most sensitive parameter is the vulnerability parameter. For SNMPv2c, this is 80% - 95%. This figure was calculated, as it is very probable an attacker can gain access to your network using default or easily guessed community strings. For SNMPv3, authPriv has been estimated at 2% - 5%. This represents the residual attack threat created by parsers at the engine level that will still need to be patched, regardless of whether you use SNMPv1, v2c, or v3. Organizations that have a greater number of assets and evidence of past breaches may also experience increased threat event frequency and exposure

factor, resulting in greater absolute dollars of loss but identical percentages of relative loss reductions. On the other hand, organizations that currently employ compensating controls (such as strict ACLs and out-of-band management networks) would likely experience reduced starting vulnerability values for SNMPv2c, reducing the relative loss reductions.

Table 4: FAIR Risk Model Assumptions and Parameters

Parameter	SNMPv2c Value	SNMPv3 authPriv Value
Threat Event Frequency	12 to 52 events/year	12 to 52 events/year
Vulnerability (Prob. of success given attempt)	80 to 95% (default/weak community)	2 to 5% (ACL + authPriv)
Annual Rate of Occurrence (ARO)	0.15 to 0.30	0.02 to 0.05
Asset Valuation (financial mainframe estate)	\$50M	\$50M
Exposure Factor (portion of asset value at risk per event)	15 to 30%	4 to 10%
Single Loss Expectancy (SLE)	\$7.5M to \$15M	\$2M to \$5M
Expected Annual Loss (EAL = ARO x SLE)	\$1.1M to \$4.5M	\$40K to \$250K
Implementation Cost (one-time)	N/A	\$50K to \$200K
Return on Security Investment (ROSI)	N/A	Payback in 1 to 3 months

6.2. Risk Reduction Visualization

The figures in Figure 4 represent the estimated annual loss for each of three different exposures. In the conservative scenario, we have used the lower end of the parameters provided in Table 4 to estimate the loss. Therefore, ARO = .015 (SNMPv2c) and ARO = .02 (SNMPv3); SLE = \$7.5 million and \$2 million; and, therefore, the EAL is \$1.1 million and \$40 thousand (a decrease of 96 percent). The middle-range values were used in the moderate scenario. Thus, ARO = .022 and .035; SLE = \$11 million and \$3.5 million; and therefore, EAL = \$2.4 million and \$123,000 (a 95% decrease). Upper bound values were used for the high-exposure scenario. Thus, ARO = .03 and .05, SLE = \$15 million and \$5 million, and therefore EAL = \$4.5 million and \$250,000 (a 94% decrease). Regardless of which of these three exposure scenarios you use to calculate your estimated annual loss (EAL) when comparing SNMPv2c to SNMPv3 authPriv, there will be a reduction in EAL ranging from 90 to 96%.

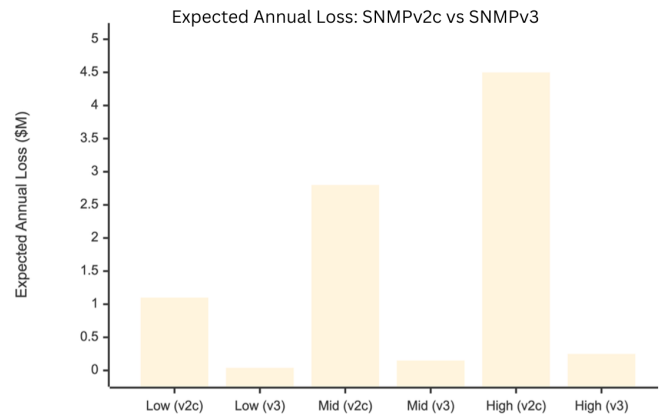


Figure 4: Expected Annual Loss before and after SNMPv3 migration across three exposure scenarios. Reductions of 90 to 96 percent are achieved through the combination of credential elimination, AES encryption, and ACL-enforced access restriction.

6.3. Compliance Cost Avoidance

In addition to direct breach cost reduction, migration eliminates material compliance liabilities. PCI-DSS v4.0 Requirement 2.2.7 [14] mandates that all non-console administrative access use strong cryptography. Auditor findings of plaintext SNMP in cardholder data environment (CDE) adjacent segments can result in non-compliance penalties of up to \$100,000 per month per card brand. SOX Section 404 requires that the internal control environment over financial reporting systems be assessed annually; insecure SNMP on mainframe infrastructure contributes to material weakness findings. NIST SP 800-53 [15] Rev. 5 control SC-8 (Transmission Confidentiality and Integrity) is directly violated by SNMPv2c and satisfied by SNMPv3 authPriv.

In addition to direct breach cost reduction, migration eliminates material compliance liabilities. PCI-DSS v4.0 Requirement 2.2.7 [14] mandates that all non-console administrative access use strong cryptography. Auditor findings of plaintext SNMP in cardholder data environment (CDE) adjacent segments can result in non-compliance penalties of up to \$100,000 per month per card brand. SOX Section 404 requires that the internal control environment over financial reporting systems be assessed annually; insecure SNMP on mainframe infrastructure contributes to material weakness findings. NIST SP 800-53 [15] Rev. 5 control SC-8 (Transmission Confidentiality and Integrity) is directly violated by SNMPv2c and satisfied by SNMPv3 authPriv.

7. Migration Methodology

This five-step plan allows SNMPv2c and SNMPv3 to live side by side with no service disruptions, as shown in Figure 5.

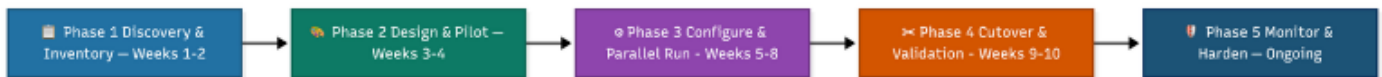


Figure 5: Five-Phase SNMPv2-to-SNMPv3 Migration Workflow from asset discovery through ongoing hardening.

7.1. A. Phase 1: Discovery and Inventory (Weeks 1-2)

The discovery phase finds all SNMP-enabled systems in your network. Device agents may be discovered using a variety of automated tools, such as nmap, scanning them at TCP/UDP port 161, and attempting community string enumeration [16]. Possible outputs of the discovery phase are device host name, host IP address, operating system type and version, community strings in use, the Net Manager System that is doing the polling (if any), trap receivers, user-created MIBs, and SNMP version. Note that some older z/OS releases and devices have no support for SNMPv3. For instance, IBM z/OS releases prior to 2.3 will require an upgrade in order to enable SNMPv3. Cisco IOS versions prior to 12.3(4)T have no native support for SNMPv3 (with aes-128). This software will have to be upgraded or removed prior to migrating your network to the next evolution level. For z/OS: IBM z/OS 2.3 and beyond can support SNMPv3 AES provided ICSF (Integrated Cryptographic Service Facility) is online.

7.2. Phase 2: Design and Pilot (Weeks 3-4)

We came up with the credential naming scheme, the VACM view structure, and the testing procedures during the design phase. The VACM view architecture splits access for monitoring (read-only) from access for management (write-capable). Users who can only read can only see the system, interfaces, IP, and TCP MIB subtrees. The writeView is empty for everyone except for a "break-glass" account that can only be used during scheduled maintenance windows. Pilot deployments included five to ten devices: one z/OS LPAR, one Cisco IOS router, and three Cisco IOS switches. They all ran SNMPv3 alongside SNMPv2c credentials that could be used as a backup.

VACM validation during the pilot utilized "snmpwalk -v3" using each of the user credentials for USM to verify that only specific OIDs within their respective subtree views could be accessed. If a "snmpwalk" were performed outside of the specified subtree view(s), it would have returned either "noSuchObject" or "endOfMIBView," verifying that VACM was enforcing its policies on the agent. The writeView restrictions were tested by performing an SNMP SET operation utilizing the read-only monitoring user. This operation resulted in a "no access" error being generated. Each VACM denied event was documented as a SERVAUTH violation in the z/OS OSNMPD job log for routing into the SIEM for documentation prior to going live.

7.3. Phase 3: Configuration and Parallel Run (Weeks 5-8)

The configuration steps are run tier by tier within the network management platform. IBM z/OS adds user IDs (the USM_USER [17] option) for all z/OS SNMP management accounts. Includes the VACM group memberships (VACM_GROUP) that define the VACM subtrees (VACM_VIEW) and links them with VACM access entries (VACM_ACCESS), specifying security requirements in terms of the authPriv security level. Conserves COMMUNITY entries for the polling mechanism being supported throughout the parallel run of the implementation.

A representative z/OS SNMPD.CONF [18] configuration for a financial mainframe SNMPv3 deployment is as follows:

```

USM_USER nmsMon01 - HMAC-SHA <40-hex-auth-key>
AESCFB128 <32-hex-priv-key> L nonVolatile
VACM_GROUP monGroup USM nmsMon01 -
VACM_VIEW monView system - included -
VACM_VIEW monView interfaces - included -
VACM_VIEW monView ip - included -
VACM_VIEW monView tcp - included -
VACM_ACCESS monGroup - - AuthPriv USM monView -
monView -
COMMUNITY readOnly public noAuthNoPriv
Localized authentication and privacy keys are generated
using the pwtokey utility:
pwtokey -p HMAC-SHA -u auth <authPassword>
<hostname_or_engineID>
pwtokey -p HMAC-SHA -u priv <privPassword>
<hostname_or_engineID>
  
```

The corresponding Cisco IOS configuration [19] is:

```

ip access-list standard SNMP-ALLOW
permit 10.0.100.50 !NMS primary
permit 10.0.100.51 !NMS secondary
deny any log
snmp-server view MonView iso included
snmp-server group MonGroup v3 priv read MonView access
SNMP-ALLOW
snmp-server user nmsMon01 MonGroup v3 auth sha AuthP4ss!
priv aes 128 Pr1vP4ss!
snmp-server host 10.0.100.50 traps version 3 priv nmsMon01
snmp-server community tempReadOnly RO SNMP-ALLOW
  
```

Cross-platform authentication for validating keys has confirmed that the z/OS platform and the Cisco IOS-XE platform are correctly authenticated with their respective USM (User Security Model) credentials. On the z/OS side, you must first use D ICSF,STATUS to validate that AES-128 hardware acceleration is enabled and then perform a test SNMP GET using the localized key and a corresponding SNMP GET request from an external network manager. The test should confirm a valid

response from OSNMPD. On the Cisco IOS-XE side, you must display the SNMP user information to confirm the auth and priv protocol being used. Both platforms have been validated to work together using the same USM username and thus confirmed to provide symmetric HMAC-SHA-96 validation at both ends of the management path.

For the receipt of NMS traps, the default behavior of z/OS SNMPD is to create its snmpEngineID from its hostname, while the default behavior of Cisco IOS-XE is to create its engineID based on the chassis's MAC address. As such, the engineIDs created on each platform must be registered in advance with the NMS prior to receiving SNMPv3 traps. Inform PDUs are recommended when migrating to SNMPv3, as they are acknowledged and thus eliminate the silent failure condition resulting from a mismatch of engineIDs between the NMS and the target system.

RACF provides two levels of integration with SNMPv3 access control functions on z/OS. First, RACF requires that a SERVAUTH class profile of type EZB.SNMPAGENT.sysname.tcprocname be defined and that the OSNMPD userid be granted permission before a DPI connection can successfully occur. If no such profile exists or if no such permissions exist, OSNMPD will terminate with an RACF violation, and subagent registration will silently fail. Secondly, RACF may prevent unauthorized operators from modifying SNMPD.CONF and restarting OSNMPD by protecting the USS path associated with SNMPD.CONF and the started task profile of OSNMPD. In addition, all SNMPv3 USM user entries and VACM group membership definitions in SNMPD.CONF should be version controlled and restricted from unauthorized modification so that a RACF violation and a SIEM event will be generated before any modification occurs to a running agent.

7.4. Phase 4: Cutover and Validation (Weeks 9-10)

Cutover removes all SNMPv2c community strings from each device immediately after confirming that all NMS polling and trap reception is operating correctly via SNMPv3. On Cisco devices [20], each community string removal also requires removing the auto-generated v1 and v2c SNMP groups. On z/OS, all COMMUNITY entries are removed from SNMPD.CONF and OSNMPD is recycled with a warm restart. Validation confirms that the show snmp community on Cisco returns an empty output, that the z/OS OSNMPD job log shows no version 2c OID walk activity, and that all NMS dashboards show data continuity without gaps. Pre-defined rollback triggers suspend cutover automatically: more than 5 percent of devices unreachable after SNMPv2c removal, NMS data gaps exceeding 15 minutes, or trap delivery failure from any production device.

The first thing that happens when a rollback trigger occurs is to "turn back" COMMUNITY entries on the impacted device tier. For z/OS, this includes restoring the COMMUNITY lines to SNMPD.CONF and doing a warm restart of OSNMPD. All the SNMPv3 USM_USER entries for each device will remain unchanged. On Cisco IOS-XE, you simply add the SNMP-server community directive again but with a restriction to the SNMP-ALLOW ACL. During rollback, none of the SNMPv3 SNMP-server user/group configurations are changed or deleted. Before attempting another cutover, change management must conduct a Root Cause Analysis (RCA) gate check, documenting the trigger, impacted devices, and corrective actions in the change management ticket. This should be recorded as a specific runbook step within your change management system.

7.5. Phase 5: Ongoing Monitoring and Hardening (Post-Cutover)

After cutover, monthly credential rotation should be scheduled for all USM users using the pwtokey and SNMP-USER-BASED-SM-MIB keyChange mechanism. Quarterly SNMP scanning with nmap confirms that UDP port 161 is not reachable from outside the management VLAN. VACM view users should be reviewed every year or when access profiles are modified. Each OSNMPD session that issues RACF permission violations via SNMP is sent to a centralized Security Information and Event Management (SIEM) system for analysis and alerting.

7.6. Deployment Observations

The five phases of methodology mentioned throughout the paper are being used in an actual production finance mainframe environment running IBM z/OS version 2.5 and utilizing both Cisco IOS-XE network boundary devices and SolarWinds as our primary Network Management System (NMS). As such, we have identified several qualitative operational deployment characteristics from that use case. In the first instance, Phase Three (the Parallel Coexistence Phase), there were no service disruptions generated; SNMPv3 and SNMPv2c polling ran concurrently for 48 hours, with no metric discontinuities reported in our NMS dashboard. This validated the coexistence method prior to removing any community strings. Secondly, the Engine ID Synchronization issue illustrated in Table 5 occurred when executing a pilot run of the z/OS LPAR environment: a test LPAR clone invalidated the locally defined USM keys. Thus, it is necessary to synchronize Engine IDs prior to generating keys, as stated in Section 8. Lastly, RACF SERVAUTH violations experienced upon initiating OSNMPD startup on two LPARs were eliminated once the EZB.SNMPAGENT profiles were created according to Section 7C. Our baseline number of violations fell to zero in 24 hours after the cutover date, which established a

clean baseline for future anomaly detection. Finally, none of the rollback triggers activated during production cutover, and all of our NMS dashboards continued to show continuous data flow without gaps through legacy community string elimination. These findings support the phasing approach and verify the checklist items listed in Table 8 as valid gates for successful production deployment.

8. Implementation Challenges and Mitigations

Engine ID management is among the most operationally critical prerequisites for a successful implementation of SNMPv3. The engine ID is identified as part of the first time the application is deployed by running `snmpget -v3 -u "" -l noAuthNoPriv 1.3.6.1.6.3.10.2.1.1.0`. For this reason, the engine ID must be recorded along with the device in your Configuration Management Database (CMDB) prior to generating localized keys since the process used to derive these keys is dependent upon the engine ID (see RFC 3414 section 2.6). In the event that an engine ID has been modified due to LPAR cloning or SNMPD.BOOTs file corruption on z/OS, then it will invalidate all previously generated User-based Security Model (USM) keys. These must be recreated by executing `pwtokey` against the new engine ID value. The steps to recreate keys are as follows: (1) Identify the new engine ID from either D ICSF,STATUS, or SNMPD.BOOTs; (2) execute `pwtokey -p HMAC-SHA -u auth` for each of the defined USM users; (3) modify SNMPD.CONF to include the new hexadecimal string representations of the localized keys; (4) perform a warm

restart of OSNMPD; and (5) update the credential records within your NMS for both the authentication and privacy keys. Track your engine IDs in your configuration management database just as you would track IP addresses.

The reality is SNMPv3 migrations almost never fail on a technical level. What causes trouble are all the things that have to be in place before they start or can happen after they start. The seven problems in Table 5, as well as the key to their successful execution in a financial mainframe migration, lie in these environmental preconditions. [21], [22].

SNMPv3 represents a structural defense against an internal (insider) threat as it does not provide an insider threat capability as does SNMPv2c. Anybody who knows the SNMPv2c community string will have full SNMP access to all devices they are allowed to view, without the ability to be held accountable individually; e.g., every poll, set, etc., can be attributed to the community string only and NOT to a particular user. In SNMPv2c an attacker can perform a full MIB tree query to gather ALL of the network topology information from a managed device(s), modify device configurations, and/or steal network topology information without ANY type of audit trail being generated to link these actions back to the actual user performing them. The SNMPv3 user-based security model provides per-user credentials with localized keys, such that each SNMP operation performed by the user is cryptographically bound to a specific username.

Table 5: Implementation Challenges and Recommended Mitigations

Challenge	Root Cause	Recommended Mitigation
engineID mismatch on z/OS	SNMPD.BOOTs file corruption or LPAR cloning changes engineID, invalidating localized keys	Store engineID from SNMPD.BOOTs in configuration management; regenerate keys after any engineID change
AES not available on z/OS	ICSF is not active or CEX hardware is not configured	Activate ICSF SSDD; verify with D ICSF,STATUS; fall back to DES only as temporary measure
NMS SNMPv3 trap reception failure	SNMPv3 traps bind to the sending agent engine ID; NMS needs createUser entry per source	Pre-configure NMS with all source engine IDs; use INFORM PDUs (acknowledged) instead of TRAPs
Legacy devices without SNMPv3 support	Firmware too old; hardware end-of-life.	Segment legacy devices in isolated management VLAN with strict ACLs; schedule decommission
Administrator training overhead	SNMPv3 configuration requires understanding of USM, VACM, key localization, and engine IDs	Produce internal runbooks with copy-paste configuration templates; conduct lab exercises before production rollout
RACF SERVAUTH violations on OSNMPD	Subagent DPI connections require SERVAUTH class profiles not present in default RACF configuration	Define EZB.SNMPAGENT.sysname.tcpprocname profiles in SERVAUTH class; permit OSNMPD userid
SolarWinds node re-discovery	Changing SNMP version resets some performance baseline counters	Export baseline data before cutover; use Orion SDK to update credentials without deleting the node

The View-Based Access Control Model (VACM) then limits the amount of the MIB subtree that a given user has access to based upon their role within the organization; i.e., a monitoring operator would not be able to issue any set operations, and there could be a separate audit trail generated when using a break-glass account. When combined with the RACF SERVAUTH enforcement capabilities provided by IBM's z/OS operating system, each SNMP session would be attributable to a specific user ID, thereby providing individual accountability, which meets both requirements of NIST SP 800-53 Control IA-2 and PCI-DSS Requirement 8.2.1. Therefore, this per-user least-privileged access model is the primary structural defense mechanism against insider-initiated SNMP abuse in financial mainframe environments.

9. Compliance Framework Alignment

Table 6: Compliance Control Mapping for SNMPv3 Migration

Framework	Control	Requirement	SNMPv3 Satisfaction
PCI-DSS v4.0	2.2.7	All non-console admin access must use strong cryptography	authPriv with AES-128 satisfies strong cryptography requirements.
PCI-DSS v4.0	2.2	No default passwords or community strings	Community strings eliminated; USM has no defaults
PCI-DSS v4.0	8.2.1	Unique user IDs for all access	USM per-user credentials replace shared community
NIST 800-53 Rev.5	SC-8	Transmission confidentiality and integrity	HMAC-SHA-96 provides integrity; AES-128 provides confidentiality
NIST 800-53 Rev.5	IA-2	Individual identification and authentication	Per-user USM credentials with individual accountability
NIST 800-53 Rev.5	AC-3	Access enforcement	VACM enforces least-privilege MIB view access per user
NIST 800-53 Rev.5	CM-7	Least functionality; disable insecure protocols	SNMPv1/v2c disabled; SNMPv3 only enforced
NSA/CISA TA17-156A	Section 3	Use only SNMPv3 with authPriv; ACL restrict to NMS IPs	Full compliance: SNMPv3 authPriv + ACL on all devices

CIS IOS Benchmark [23]	Sec. 1.5	SNMP ACLs; encrypted passwords; no default communities	IP ACL on all SNMP groups/users; SHA passwords; communities removed [23]
------------------------	----------	--	--

10. Resource and Timeline Estimates

Table 7: Migration Task Resource Estimates (150 to 300-Device Environment)

Task	Person -Hours	Calendar Time	Key Dependencies
Asset discovery and inventory	24 to 40	1 to 2 weeks	Network access; NMS query permissions
VACM view design and credential schema	16 to 24	1 week	RACF engineering; NMS admin; security architect
Pilot deployment and validation	20 to 32	1 to 2 weeks	5-10 representative devices; change window
Configuration automation scripting	32 to 48	1 to 2 weeks	Python/Ansible ; device reachability; lab environment
Phased rollout (150-300 devices)	80 to 120	3 to 4 weeks	Change management; maintenance windows
NMS reconfiguration (SolarWinds/Zabbix)	16 to 24	1 week	Parallel with device rollout
Cutover and legacy removal	12 to 20	3 to 5 days	Rollback plan: monitoring continuity confirmed
Post-migration validation and documentation	16 to 24	1 week	Compliance evidence package; runbook finalization
TOTAL (excluding ongoing hardening)	216 to 332	9 to 11 weeks	

11. Repeatable Playbook and Verification Checklist

The checklist in Table 8 encapsulates the complete migration verification sequence. Each item maps to a specific phase and can be independently audited as

evidence for PCI-DSS and NIST compliance assessments. Teams should use this checklist as a gate between phases: no phase advances until all preceding items are checked.

Table 8: SNMPv2-to-SNMPv3 Migration Verification Checklist

#	Verification Item	Phase	Compliance Ref.
1	All SNMP-enabled assets documented with version, community strings, and NMS associations	Discovery	PCI 2.2
2	Legacy devices without SNMPv3 support identified and firmware upgrade scheduled	Discovery	CM-7
3	VACM views restrict monitoring users to required MIB subtrees only	Design	AC-3, CISA TA17-156A
4	ACLs defined restricting SNMP access to authorized NMS IP addresses only	Design	CISA TA17-156A
5	z/OS ICSF active and AES-128 confirmed available via D ICSF,STATUS	Design	RFC 3826
6	NMS v2c and v3 polling values compared; no metric discrepancies for 48 hours	Pilot	All
7	USM_USER entries with localized SHA + AES keys added to z/OS SNMPD.CONF for all users	Config	RFC 3414
8	RACF SERVAUTH profiles created for subagent DPI connections	Config	IBM SC27-3650
9	Legacy community strings removed from all devices	Cutover	PCI 2.2.7, NSA
10	All NMS dashboards confirm data continuity without gaps post-cutover	Cutover	All
11	nmap UDP 161 scan confirms SNMPv1/v2c responses no longer returned	Validation	PCI 2.2.7
12	Compliance evidence package prepared (configs, scan output, NMS screenshots)	Validation	PCI, NIST

12. Forward-Looking Considerations: Modern Telemetry

SNMPv3 is the mandatory remedy, especially for the existing fleet of financial mainframe systems. The telemetry landscape is changing, with gNMI [24] (streaming over Protocol Buffers/HTTP/2 with mandatory TLS) and NETCONF [25] (SSH-encrypted YANG-modeled configuration) [26] becoming more popular. Neither IBM z/OS Systems nor IBM Network Switch and Gateway Services presently understands gNMI or either NETCONF or RESTCONF. SNMP remains the only supported way to monitor your devices. The OpenTelemetry API, which IBM is implementing gradually via the Z APM Connect product, points in the right direction for network observability. Practical action in the financial-mainframe ecosystem is hybrid SNMPv3 for z/OS and OpenTelemetry for everything else, with the comforting notion that SNMP in the z/OS world is unlikely to go away anytime in the next decade.

This paper specifically aligns its migration strategy with NSA/CISA critical infrastructure guidelines. Both the December 2024 CISA Enhanced Visibility and Hardening Guidance [11] and the NSA Network Infrastructure Security Guide [3] require SNMPv3 be used by all critical infrastructure sectors. Since financial mainframes are classified as critical infrastructure under CISA's Financial Services Sector designation, the above guidance applies directly and therefore is an acceptable position to defend against regulatory examination and incident investigation inquiries. The five-stage migration process and VACM least privilege configuration described in Section 7, combined with the ACL-restricted MGMT plane and Engine ID lifecycle control described in Section 8, provide network hardening that meets or exceeds the guidelines contained in both the CISA and NSA documents.

13. Limitations and Future Work

13.1. Limitations

The FAIR [13] risk reduction figures are based upon the assumptions below for Threat Event Frequency, Vulnerability, Exposure Factors, and Asset Value. These assumptions use a mid-sized financial institution having 150-300 SNMP-enabled devices as a base case. Note that the absolute reduction figures derived from this case study will differ between financial institutions of a different size. Nonetheless, the model suggests that a 90-96% reduction can be expected regardless of the size of the institution under consideration. There is no post-migration actual data available for specific migration projects.

The procedures to migrate from IBM z/OS 2.5 and Cisco IOS-XE are described in full detail. Migration from other vendors, such as Junos OS from Juniper, EOS on Arista switches, F5 BIG-IP load balancers, and storage

controllers, is touched on at the surface. It is still a requirement to remedy engine-level parser vulnerabilities such as CVE-2017-6742 [7] and CVE-2025-20352 [27], affecting both SNMPv2c and SNMPv3 modules. HMAC-SHA-96 per RFC 3414 is the specified baseline because HMAC-SHA-2 variants defined in RFC 7860 are not generally supported on z/OS or Cisco IOS-XE trains. This paper does not include measurements for the processor, memory, and communication overhead of using SNMPv3 in an LPAR on z/OS.

13.2. Future Work

SNMPv3 testing also would be beneficial for vendors such as Juniper, Arista, F5, and various storage controller vendors. As SNMPv3 and network devices more widely support stronger authenticator algorithms, such as HMAC-256 recommended in RFC 7860, their adoption becomes viable. Architecture designs that employ hybrid data collection schemes between SNMPv3 collection from mainframe z/OS devices and the gNMI [24], NETCONF [25], and SNMPv3-supported OpenTelemetry provide additional architectural guidance to designers. In the same spirit as the previous statement, there also exist automation software such as Ansible and tools such as pyATS that significantly reduce the effort required to operate, validate, and roll back during the migration of large heterogeneous SNMP fleet systems, which require only a few percent of the person-hour figures previously reported in Table 7 to perform such large migration projects. Longitudinal observational studies on network systems affected by such upgrades or rollbacks also can provide additional useful information to designers.

Empirical benchmarks of SNMPv3 authPriv CPU and memory usage should be conducted on z/OS LPARs that use ICSF hardware-accelerated crypto rather than software-only crypto to create the type of quantitative information that architects require when deciding which LPAR resource allocations will meet their needs after migrating their current systems. In addition to validating the results of the benchmarks, these tests would also serve as a testbed to determine if the high CPU utilization noted in the Limitations section can be mitigated solely through the use of hardware cryptography or if some reduction in polling frequency will still be required. If an organization chooses to leverage configuration management automation (such as Ansible) with either IBM z/OS-compatible modules or PyATS for Cisco IOS-XE validation, they will potentially reduce the amount of time spent on each device by 60-80%, as indicated in the 3rd row of Table 7, for example.

AI-based anomaly detection can be applied to SNMPv3 traps and inform and alert an administrator whenever there is anomalous behavior detected (e.g., an unauthorized polling attempt or a sudden change in engine ID or a violation of VACM policies). This method

provides an advantage over traditional static SIEM correlation rules because it has the capability to detect anomalies as they occur in near-real-time, whereas static SIEM correlation rules rely upon historical knowledge. Automating Configuration Deployment and Using AI-Assisted Post-Migration Monitoring represent the natural next iteration of the Repeatable Playbook described above. To better understand how to scale the SNMPv3 management plane to accommodate cloud-resident network devices and containerized workloads, the scalability of SNMPv3 to accommodate hybrid cloud models requires further research.

14. Conclusion

This paper has demonstrated that SNMPv2c is fundamentally flawed as a mechanism for remotely managing financially critical mainframes. It has also demonstrated that a fully qualified path to securely remotely access such systems exists, is proven to work in production, and is in the economic reach of even the most fiscally lean enterprise. To address these flaws, SNMPv3 provides security using the User-based Security Model, defined by RFC 3414. Instead of using a simple community string for plaintext communication, SNMPv3 provides cryptographic authentication of host identity via HMAC-SHA-96 and optional confidential communication via AES-128 CFB. The SNMP view-based access control model defined in RFC 3415 enables granular, auditable control over SNMP access.

The FAIR-based risk analysis establishes that migration reduces expected annual loss from the \$1.1M to \$4.5M range under SNMPv2c to \$40,000 to \$250,000 under SNMPv3 authPriv, a reduction of 90 to 96 percent. This calculation does not include compliance penalty avoidance or reputation protection value, making the actual benefit larger. The migration investment of \$50,000 to \$200,000 and nine to eleven weeks of elapsed time is recovered within one to three months of post-migration operation.

The five-phase methodology presented here, from structured inventory through parallel-run validation and legacy removal, generalizes to any organization operating SNMP-managed infrastructure. The parallel coexistence capability of both z/OS SNMPD.CONF and Cisco IOS eliminates the primary operational risk of migration. The 12-item verification checklist provides an independent, audit-ready evidence record. For security architects and mainframe operations teams, the conclusion is unambiguous: SNMPv2c should be disabled immediately, and SNMPv3 with authPriv, SHA, AES-128, and ACL-restricted access to authorized NMS hosts should be the only SNMP version operating on financial mainframe infrastructure.

Funding

This research received no external funding.

Author Contributions

Rohit Kumar Shaw is the sole author and conducted all aspects of this research, including conceptualization, methodology, analysis, and writing.

Conflicts of Interest

The author declares no conflict of interest.

References

- [1] J. Case, R. Mundy, D. Partain, and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework," IETF RFC 3410, Dec. 2002, doi: <https://doi.org/10.17487/RFC3410>.
- [2] Cybersecurity and Infrastructure Security Agency, "Reducing the Risk of SNMP Abuse," Alert TA17-156A, Jun. 5, 2017. [Online]. Available: <https://www.cisa.gov/news-events/alerts/2017/06/05/reducing-risk-snmp-abuse>
- [3] National Security Agency, "Network Infrastructure Security Guide," Cybersecurity Technical Report U/OO/118623-22, ver. 1.2, Oct. 2023. [Online]. Available: <https://media.defense.gov>
- [4] D. Harrington, R. Presuhn, and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks," IETF RFC 3411, Dec. 2002, doi: <https://doi.org/10.17487/RFC3411>.
- [5] U. Blumenthal and B. Wijnen, "User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)," IETF RFC 3414, Dec. 2002, doi: <https://doi.org/10.17487/RFC3414>.
- [6] B. Wijnen, R. Presuhn, and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)," IETF RFC 3415, Dec. 2002, doi: <https://doi.org/10.17487/RFC3415>.
- [7] National Vulnerability Database, "CVE-2017-6742 Detail: Cisco IOS SNMP Buffer Overflow," 2017. [Online]. Available: <https://nvd.nist.gov/vuln/detail/CVE-2017-6742>
- [8] National Cyber Security Centre, National Security Agency, Cybersecurity and Infrastructure Security Agency, and Federal Bureau of Investigation, "APT28 Exploits Known Vulnerability to Carry Out Reconnaissance and Deploy Malware on Cisco Routers," Joint Advisory, Apr. 2023. [Online]. Available: <https://www.ncsc.gov.uk>
- [9] U. Blumenthal, F. Maino, and K. McCloghrie, "The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model," IETF RFC 3826, Jun. 2004, doi: <https://doi.org/10.17487/RFC3826>.
- [10] N. Lawrence and P. Traynor, "Under New Management: Practical Attacks on SNMPv3," in Proc. 6th USENIX Workshop on Offensive Technologies (WOOT), Bellevue, WA, USA, Aug. 2012, doi: <https://doi.org/10.5555/2372399.2372416>.
- [11] Cybersecurity and Infrastructure Security Agency, "Enhanced Visibility and Hardening Guidance for Communications Infrastructure," Dec. 2024. [Online]. Available: <https://www.cisa.gov>
- [12] Cisco Systems, "Cisco IOS and IOS XE Software SNMP Denial of Service and Remote Code Execution Vulnerability," Security Advisory cisco-sa-snmp-x4LPhte, Sep. 24, 2025. [Online]. Available: <https://sec.cloudapps.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-snmp-x4LPhte>
- [13] J. Freund and J. Jones, Measuring and Managing Information Risk: A FAIR Approach. Waltham, MA, USA: Butterworth-Heinemann, 2014.
- [14] PCI Security Standards Council, "Payment Card Industry Data Security Standard: Requirements and Testing Procedures, Version 4.0," Mar. 2022. [Online]. Available: <https://www.pcisecuritystandards.org>
- [15] Joint Task Force, "Security and Privacy Controls for Information Systems and Organizations," National Institute of Standards and Technology, Gaithersburg, MD, USA, NIST Special Publication 800-53 Rev. 5, Sep. 2020, doi: <https://doi.org/10.6028/NIST.SP.800-53r5>.
- [16] J. Schönwälder, A. Pras, M. Harvan, J. Schippers, and R. van de Meent, "SNMP Traffic Analysis: Approaches, Tools, and First Results," in Proc. 10th IFIP/IEEE Int. Symp. Integrated Network Management (IM), Munich, Germany, May 2007, pp. 323–332, doi: <https://doi.org/10.1109/INM.2007.374797>.
- [17] IBM, "USM_USER Entry," IBM z/OS V2.5.0 Documentation. [Online]. Available: <https://www.ibm.com/docs/en/zos/2.5.0>
- [18] IBM, "SNMPD.CONF Sample," IBM z/OS V2.5.0 Documentation. [Online]. Available: <https://www.ibm.com/docs/en/zos/2.5.0>
- [19] Cisco Systems, "Configuration Template for SNMPv3," Cisco Community Knowledge Base, Document ID 4666450. [Online]. Available: <https://community.cisco.com>
- [20] Cisco Systems, "Disable SNMPv1 or SNMPv2c While Other Versions Remain Enabled," Document ID 113469. [Online]. Available: <https://www.cisco.com>
- [21] L. Andrey, O. Festor, A. Lahmadi, A. Pras, and J. Schönwälder, "Survey of SNMP Performance Analysis Studies," International Journal of Network Management, vol. 19, no. 6, pp. 527–548, Nov. 2009, doi: <https://doi.org/10.1002/nem.729>.
- [22] S. Kuryla and J. Schönwälder, "Evaluation of the Resource Requirements of SNMP Agents on Constrained Devices," in Managing the Dynamics of Networks and Services, Lecture Notes in Computer Science, vol. 6734. Berlin, Germany: Springer, 2011, pp. 100–111, doi: https://doi.org/10.1007/978-3-642-21484-4_13.
- [23] Center for Internet Security, "CIS Cisco IOS Benchmark," ver. 2.2, 2023. [Online]. Available: <https://www.cisecurity.org>
- [24] OpenConfig, "gNMI Specification," ver. 0.10.0, GitHub, 2023. [Online]. Available: <https://github.com/openconfig/gnmi>
- [25] R. Enns, M. Björklund, J. Schönwälder, and A. Bierman, "Network Configuration Protocol (NETCONF)," IETF RFC 6241, Jun. 2011, doi: <https://doi.org/10.17487/RFC6241>.
- [26] F. K. Ariefputra and E. Mulyana, "Performance Analysis of gNMI Streaming Telemetry-Based Monitoring Systems Using Containerlab Network Simulation," Jurnal Nasional Teknik Elektro dan Teknologi Informasi, vol. 13, no. 2, pp. 101–107, 2024, doi: <https://doi.org/10.22146/jnteti.v13i2.10185>.
- [27] National Vulnerability Database, "CVE-2025-68615 Detail: Net-SNMP snmptrapd Buffer Overflow," 2025. [Online]. Available: <https://nvd.nist.gov/vuln/detail/CVE-2025-68615>

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).

Rohit Kumar Shaw is an Infrastructure Engineer specializing in mainframe infrastructure and network security with deep expertise in IBM z/OS, z/VM, SNMP, AT-TLS, RACF and ACF2, NetView, Omegamon, Splunk, and z/OSMF. He holds a degree in Computer Science and has worked across disaster recovery testing, certificate renewals, SNMPv2-to-SNMPv3 hardening, AT-TLS policy

deployment, TCP/IP configuration standardization, and platform modernization in financial services environments. His current research interests include zero-trust transport security on legacy mainframes, SSH compliance automation under z/OS USS, RAG-based retrieval for enterprise operations, and Kerberos versus public key authentication patterns. He has written extensively on these topics for IEEE-format research papers and trade publications, with a focus on producing publication-ready guidance for security architects and mainframe operations teams.

Beyond Written Surveys: Validating Voice-Based Implementations of the User Experience Questionnaire

Ignacio Diaz-Oreiro*, Gustavo Lopez

School of Computer Science and Informatics, University of Costa Rica, San Jose, 11501, Costa Rica

Email(s): gustavo.lopezherrera@ucr.ac.cr (G. Lopez)

*Corresponding author: Ignacio Diaz-Oreiro, San Jose, Costa Rica +50625118000 & Email: ignacio.diazoreiro@ucr.ac.cr

ABSTRACT: User Experience (UX) evaluation is fundamental for digital product improvement, yet traditional written questionnaires face limitations in engagement, accessibility, and response consistency. To address this, we present the design, development, and validation of voice-based adaptations of the User Experience Questionnaire, or UEQ, using natural conversational interfaces. This research introduces two distinct implementations: direct scale mapping approach called Voice1-7, and a novel branched-dialog approach called Voice2Q, which uses sequential questions to capture attitude direction and intensity. Further, we propose Voice2Q+VC, a voice-first variant with minimal visual cues to enhance clarity while preserving voice interaction primacy. Multi-stage validation was conducted through multiple case studies involving 345 participants in the design and evaluation phases. These studies compared voice implementations against the standard written version of UEQ. Statistical analyses across diverse participant groups confirmed that both Voice2Q and Voice1-7 produced similar results to the written UEQ in core UX dimensions —Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, Novelty— establishing their measurement validity. A Usability, UX and cognitive workload comparison was conducted. Results revealed that Voice2Q significantly reduced response inconsistencies and excelled in UX hedonic stimulation, while the written UEQ retained advantages in UX pragmatic efficiency. Voice2Q+VC mitigated workload challenges inherent in voice-only interactions, outperforming the written UEQ in Usability and UX hedonic dimensions while approaching its UX pragmatic performance levels. These results suggest that supplemental visuals can optimize voice-driven evaluation without sacrificing conversational engagement. This work confirms voice-based UEQ as a statistically valid, accessible alternative to written formats, with benefits in response reliability and user engagement. The Voice2Q+VC implementation presents a promising paradigm for balancing natural interaction with cognitive efficiency in UX evaluation.

KEYWORDS: User Experience, Voice Interfaces, User Experience Questionnaire, Human-Computer Interaction, Usability, NASA-TLX, System Usability Scale (SUS), Natural Conversation Framework (NCF)

1. Introduction

User Experience (UX) is critical for the success of digital products and services. To enhance UX and user satisfaction, it is essential to understand how users perceive and interact with products. One of the most common ways to evaluate UX are standardized questionnaires. However, filling out a written questionnaire can be tedious and prone to inconsistencies, affecting the quality and reliability of collected data [1].

Voice-based interfaces could provide more natural and intuitive interaction, which could also take advantage of the widespread availability of smartphones and smart speakers. The voice approach offers several benefits and its positive effects have been already described, including increased accessibility for users with disabilities or low literacy, reduced inconsistencies for better response rates, and natural and conversational interactions, providing more accurate and meaningful feedback from users [2, 3].

This paper describes the development and implementation of a voice-based version of the User Experience

Questionnaire (UEQ) [4] to allow users to complete the questionnaire through voice-based interfaces. UEQ is one of the most recognized User Experience evaluation questionnaires [5] and surpasses AttrakDiff [6], the other widely used questionnaire, in number of applications reported in academic studies [7].

This research also assesses the validity and reliability of the voice-based UEQ compared to the traditional written version and presents an evaluation of the voice-based UEQ in terms of UX, Usability and workload.

The rest of the paper is structured as follows: Section 2 describes User Experience evaluation and conversational voice interfaces as background for this research. Section 3 describes the related work including the evaluation of voice interactions, how chatbots have been used in UX evaluation, the comparison of interaction modalities, and the use of semantic differential scales. Section 4 identifies the research gap and presents the motivation for this work. After that, Section 5 describes the process followed to implement our solution and Section 6 discusses the results of its evaluation. Finally, Section 7 presents our conclusions, limitations and

future work.

2. Background

This work is based on two fundamental areas: UX evaluation using standardized questionnaires and conversational voice interfaces.

2.1. User Experience evaluation questionnaires

The term User Experience was first used by Norman in the 1990s [8] to broaden the overly narrow scope of Usability and cover all aspects of a person's experience with a system [5].

There have been efforts to create a unified definition [9, 10, 11], but this is still an open topic with no consensus reached. The International Standards Organization (ISO) proposes a definition adopted in this work. According to ISO, UX encompasses a person's perceptions and responses that result from the use and/or anticipated use of a product, system or service. This includes emotions, beliefs and responses, both physical and psychological, as well as the brand image, presentation, system performance, previous experiences, attitudes, skills and personality [12].

User Experience (UX) is a key element in determining the quality of a product or service [13, 14, 15]. The goal of evaluating the UX of a product or service is to improve the user interaction [16].

An important element to consider is which method or instrument to use in UX evaluation [13, 17]. One of the most commonly used are standardized questionnaires, which contain a known and fixed set of questions to collect the end user perception of a product or service, covering both the pragmatic aspects of the interaction (clarity, ease of use or learning, predictable flows, among others) and the hedonic aspects (related to feelings, identification and stimulation) [18].

These standardized questionnaires are inexpensive and easy to administer, their use is widespread and they are considered reliable and valid for measuring UX [16, 19].

The three most recognized standardized questionnaires for UX evaluation are AttrakDiff, UEQ (User Experience Questionnaire) and mCUE (modular evaluation of the Component model of User Experience) [5, 18, 20, 21, 22].

In this research we focus on UEQ, a questionnaire that evaluates six factors: Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation and Novelty [4].

2.2. Conversational voice interfaces

The other fundamental pillar of this work is conversational voice interfaces, in which the UX is captured through natural language, made up of words and expressions [23]. These user interfaces are inspired by conversation between people [24], and they implement different levels of complexity [25].

In this work we focus on intelligent assistants, defined as software that allows automating tasks by organizing and maintaining information through oral interaction with the user [26, 27].

Now, although smart assistants have boosted the use and acceptance of voice interfaces, not all voice interactions

can be considered as a conversation. A conversation is a distinctive form of natural language use that involves particular methods of taking turns and ordering them in sequences, the persistence of context across turns, and characteristic actions to manage the interaction itself [28]. A voice command that allow a person to search for information on the Internet, or set a reminder in the form of an alarm, could not be considered a conversation.

Natural language processing has provided intelligent assistants with powerful tools to analyze spoken and written fragments of human language. However, it has not provided designers with models for how those fragments of language should be sequenced to create an interaction-like conversation, defined as a complex system of speech exchange [28]. Creating a user interface that approximates this interaction requires modeling human conversation patterns as a reflection of the complexity of the world [29], and incorporating it into computational interfaces would allow people to interact with computers by speaking naturally [30].

Designing natural language voice interfaces requires knowledge of human conversation, with its turn-taking systems, sequence organization, and repair mechanisms. Therefore, researchers in [31] created a conversational framework (NCF for Natural Conversation Framework) that presents a set of patterns that include simplified forms of natural human conversation patterns. These concepts are based particularly on the area of sociology known as Conversation Analysis, developed by sociologists Sacks [32], providing formal patterns of how people naturally speak.

3. Related Work

Related work is classified into the following sections: voice interaction evaluation, chatbots in UX evaluation, interaction modality comparison, and semantic differential scales as branching questions.

3.1. Voice interaction evaluation

Research on the evaluation of conversational systems has adopted multiple methodological approaches. These studies highlight the importance of multidimensional evaluation methodologies and standardized instruments, while also noting the limited attention given to how different dialogue types affect user expectations [33, 34, 35].

Beyond traditional evaluations, conversational interfaces have also been explored in specific application domains such as accessibility and healthcare. Prior work has shown that conversational user interfaces can improve accessibility for users with disabilities [36], while studies comparing voice and written responses report comparable results and support the feasibility of voice-based questionnaires and surveys [37, 38].

Additionally, approaches such as Kansei engineering have explored the measurement of emotional satisfaction in voice-based intelligent systems through conversational design parameters, identifying their influence on dimensions such as pleasure and reliability [39].

3.2. Chatbots in UX evaluation

Regarding chatbots, prior research has explored their use in UX evaluation and survey administration. In [40], authors investigated the use of chatbots as assistants for UX evaluators through the Wizard of Oz technique, laying the groundwork for collaborative tools between humans and AI in UX evaluation

Studies comparing conversational approaches with traditional web forms report mixed results. While some research found no significant advantages of chatbots over web forms in terms of enjoyment and usefulness [41]. Other studies reported higher response rates, better data quality, and lower straight-lining in chatbot-based surveys [42]. Additionally, conversational questionnaires have shown positive user perceptions in healthcare contexts, where participants preferred chatbot-based assessments despite longer completion times [43].

3.3. Interaction modality comparison

In [44], authors assessed the validity and reliability of data collected through a Google Assistant application using yes/no questions and Likert scales answered either by voice or through screen interaction. Results revealed comparable internal consistency and validity between both mechanisms.

Similarly, in [45], researchers developed a conversational survey tool using a chat-like web interface with traditional written controls such as buttons, sliders, and multiple-choice questions. Researchers conclude that users appreciate the conversational form and prefer it over traditional approaches.

Regarding conversational voice interfaces, previous studies in healthcare contexts have explored the feasibility of administering questionnaires through voice interaction. In [37], authors evaluated multiple questionnaires using both voice and written modalities, finding that although oral versions were perceived as more difficult to answer, the results did not vary significantly between modalities. Likewise, [46] compared written questionnaires with voice interfaces using discrete and open responses, reporting higher correlations between written responses and voice interactions based on discrete numerical answers.

3.4. Semantic differential scales as branching questions

Finally, it is also worth mentioning related work to implementing semantic differential scales through two successive questions, also called branching questions, since this technique is part of the voice questionnaire that will be presented in Section 4.

Previous studies on telephone surveys explored presenting attitude questions in two stages: first identifying the direction of the attitude and then classifying its intensity [47, 48]. This branching approach was proposed to address the difficulty of administering semantic differential scales without visual support. The term branching questions represents that the wording of the follow-up question varies depending on the initial response provided.

Research on political preference surveys reported that breaking down the reporting process into two steps improves the speed, ease, reliability, and predictive validity of

responses [1]. Similarly, studies comparing traditional one-stage semantic differential scales with two-stage branching formats found that branching questions generated a higher percentage of extreme-position responses, suggesting lower central tendency bias [49].

More recently, branching questions have also been studied in telephone surveys to analyze whether dividing a question into several stages reduces unanswered or undefined responses [50]. However, these studies were conducted through human-administered telephone interactions, which may affect participants' responses compared to conversational systems or self-reported voice-based questionnaires.

4. Research gap and motivation

UX is commonly evaluated through standardized questionnaires such as UEQ, AttrakDiff, SUS, and NASA-TLX, which are traditionally administered using written interfaces. At the same time, conversational systems and voice interfaces have gained relevance in multiple domains, motivating research on conversational questionnaires, chatbots, and voice-based surveys. Previous studies report promising results regarding the feasibility, validity, reliability, and user acceptance of conversational and voice-based data collection mechanisms, particularly in healthcare and accessibility contexts. Additionally, branching questions have been explored as an alternative strategy for administering semantic differential scales without visual support.

However, despite these advances, limited research has focused on implementing standardized UX evaluation instruments through conversational voice interfaces. Existing studies mainly evaluate general surveys, health-related questionnaires, or conversational forms. Little attention has been given to the voice-based implementation of semantic differential UX questionnaires such as UEQ. Therefore, this work proposes and evaluates multiple conversational voice implementations of the User Experience Questionnaire (UEQ), exploring different interaction mechanisms for administering semantic differential scales through voice interfaces.

5. Development Process

This section describes the design and implementation process of the voice-based UEQ versions, including the conversational mechanisms and interaction patterns used.

The first effort in mapping the UEQ questionnaire to a voice interface was to design an equivalent version (i.e., the questionnaire gives a 7-point scale, then the voice interfaces mapped this 7-point scale), as shown in figure 1. This implementation, called Voice1-7, is documented and described in [51].

To create an alternative version of the semantic differential, each question was implemented as shown in figure 2.

As can be seen, each item is converted into an initial question in which the direction of the attitude is defined, and a second branching question is unfolded in which the participant specifies the intensity of the attitude. If the participant indicates "neither", the second question is not

asked. Additionally, the participant could respond to both branching questions at once in a single expression. For example, saying "I think this is extremely boring" transmits both direction (boring) and intensity (extremely), applying concepts from Conversation Analysis. We called this implementation Voice2Q.

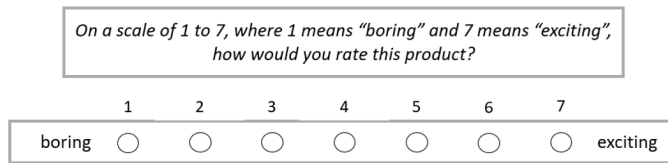


Figure 1: Direct mapping of the 7-point semantic differential scale to voice (Voice1-7). In this baseline implementation, users respond to each UEQ item by speaking a number from 1 to 7, mirroring the structure of the written questionnaire.

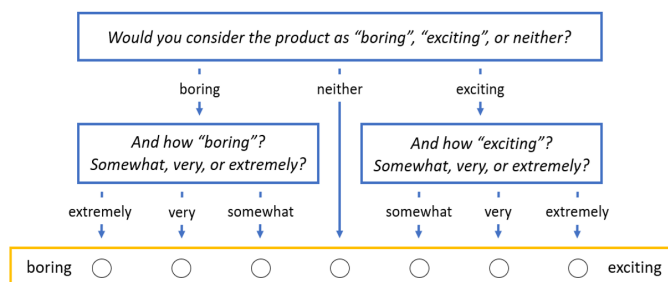


Figure 2: The branched-question approach (Voice2Q). Each UEQ item is converted into a more natural, conversational sequence: an initial question to determine the direction of the attitude, followed by a second question to gauge its intensity. The neutral option ("neither") skips the intensity question, streamlining the interaction.

Also, as part of the design of these two implementations, a Wizard of Oz exercise was conducted to compare user perceptions of both Voice2Q (branching questions) and Voice1-7 (direct 1 to 7 questions). This version only included 10 questions of the UEQ. Afterward, participants assessed the satisfaction and naturalness of both implementations and were then individually interviewed to explore their experience further. Results are discussed in Section 6.

Additionally, to explore the UX of Voice2Q, an anticipated UX evaluation was performed using a video storyboard, showing a person's interaction with the voice-implemented assessment questionnaire. The video also represented conversational elements (e.g., the ability to request a question to be repeated, correct the previous response, and request an explanation of the concepts). Results are also discussed in Section 6.

Subsequently, and based on the results of the Wizard of Oz exercise and the video storyboard evaluation, complete implementations were developed for both versions, Voice1-7 and Voice2Q, using the items from the official Spanish version of the UEQ [52].

These voice questionnaires were implemented using the VoiceFlow tool, and two case studies were conducted. The first case study, documented in [51], compared Voice1-7 with the written version of UEQ. The second case study evaluated the branched implementation (Voice2Q) and the results are shown in the Section 6. It is important to notice that Voice2Q is implemented through a set of conversa-

tional patterns shown in figure 3, based on the framework presented by [31].

As can be seen in figure 3, there are three levels in the conversational patterns. Level 1 (conversational activities) covers the implementation of the UEQ semantic differential item converted into the two questions and their variants. Level 2 (sequence management, marked B1 to B7) includes activities such as requesting the meaning of a concept or modifying a given answer. Finally, Level 3 (conversation management, marked C1 to C6) includes start and end-of-process interactions, asking the voice assistant what its skills are or where the participant is in the process of filling out the questionnaire. The full description of these patterns is presented in [53].

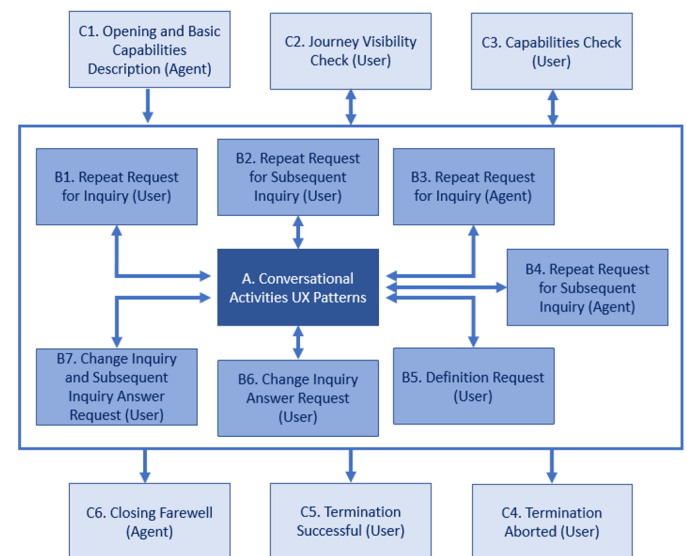


Figure 3: The conversational pattern architecture of the Voice2Q implementation, based on the Natural Conversation Framework (NCF). It organizes the interaction into three levels: Level 1 (core conversational activities for asking UEQ items), Level 2 (sequence management for repairs like repeating a question), and Level 3 (conversation management for starting and ending the session).

To explore whether visual cues could reduce workload of the voice-only implementations of Voice2Q, a new variant was developed: Voice2Q+VC, where the letters "VC" stand for "visual cues". This new version incorporates minimal visual cues only the concepts of the question, similar to the semantic differential while maintaining voice as the sole input method. As can be seen in figure 4, visual aids display the names of the concepts being evaluated and the participant's responses, mutating from a voice-only to a voice-first implementation.

The evaluation of the voice implementations was conducted across two dimensions. First, the validity of Voice2Q was assessed by comparing its results with those of the written version of UEQ using a Wilcoxon rank test, and its reliability was measured through Cronbach's Alpha coefficients. Second, the UX, Usability, and cognitive workload of Voice2Q and Voice2Q+VC were evaluated using three standardized instruments: In [6], the authors measure general UX, SUS [54] to assess Usability, and NASA-TLX [55] to quantify cognitive workload. Full results are presented in Section 6.



Figure 4: User interface concept for Voice2Q+VC (voice + visual cues). This voice-first design adds minimal, context-preserving visual aids that display the current UEQ term pair and the user's spoken response, aiming to reduce cognitive load while keeping voice as the primary and sole input method.

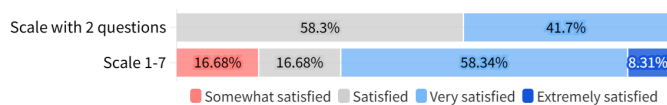
6. Evaluation and Discussion

This section presents the evaluation results across the different stages of the research, covering the Wizard of Oz exercise, the anticipated UX evaluation, and the formal validity and UX assessments of the voice implementations.

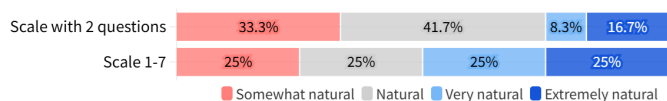
6.1. Wizard of Oz

As part of the design phase, a Wizard of Oz exercise was conducted with 12 participants (8 females and 4 males, mean age of 29.5 years) to compare Voice2Q and Voice1-7 using 10 items of the UEQ. Afterward, participants assessed the satisfaction and naturalness of both implementations and were individually interviewed. Figure 5 shows the evaluations of the two versions regarding general satisfaction, naturality, and ease of use.

How satisfied are you to express your opinion through...?



How natural was to answer questions that use...?



Questions that use this scale are easy to answer

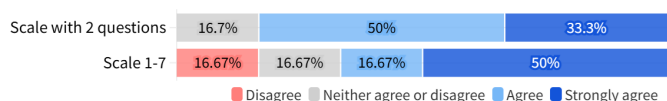


Figure 5: Results from the Wizard of Oz pilot study (n=12) comparing user perceptions of Voice1-7 and Voice2Q. While Voice1-7 was rated as simpler and easier to use, Voice2Q was preferred by participants who valued its expressiveness and conversational feel, highlighting a trade-off between simplicity and engagement.

Both implementations received similar ratings, although Voice1-7 was identified as simpler and easier to respond to. However, the implementation with two questions stood out among participants who indicated in the follow-up interview that they could give their opinion

more expressively and felt part of a conversation. This trade-off between simplicity and expressiveness motivated the continued development of Voice2Q as the primary implementation.

6.2. Anticipated UX Evaluation: Video storyboard

To explore the anticipated UX of Voice2Q, an evaluation was performed using a video storyboard showing a person interacting with the voice-implemented questionnaire. The video also represented conversational elements such as the ability to request a question to be repeated, correct a previous response, and request an explanation of concepts. The storyboard was evaluated by 197 participants using the written UEQ. However, 49 responses were considered invalid due to inconsistencies on three or more scales, leaving 148 valid responses.

Participants ranged in age from 18 to 76 years, with a median of 22 years. Of the 148 valid respondents, 71 (49.3%) identified as female, 73 (48.0%) as male, and 4 (2.7%) as other or preferred not to say. Regarding prior experience with voice technology, 47 participants reported using voice assistants or smart speakers often (31.8%), 64 sometimes (43.2%), and 37 never (25.0%).

The 148 valid responses for the video storyboard were compared against the UEQ benchmark dataset, which aggregates data from 21,175 participants across 468 studies. As can be seen in figure 6, Voice2Q obtained "Good" ratings for the Attractiveness, Perspicuity, and Novelty, and "Above average" ratings for Efficiency, Dependability and Stimulation. These results provided early evidence supporting the viability of the Voice2Q approach.

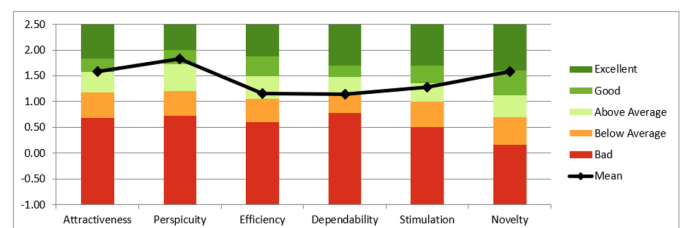


Figure 6: Anticipated UX evaluation of the Voice2Q concept using a video storyboard (n=148). The mean scores for all six UEQ scales are compared against the official UEQ benchmark dataset. Chart generated automatically using the official UEQ Data Analysis Tool [4].

In addition to the UEQ evaluation of the video storyboard, an open question was included for participants to freely express their opinion of the voice questionnaire. Of the 66 participants who provided comments, 48 were entirely positive (72.7%), 10 entirely negative (15.2%), and 8 combined both perspectives (12.2%). Positive feedback highlighted that the voice mechanism could benefit users who find reading difficult, that it allows completing a survey while doing other activities, and that it makes surveys feel more engaging and innovative. Negative comments pointed to the interaction feeling tedious or confusing, and to the assistant's voice in the video could be more humane and pleasant.

6.3. Validity evaluation

Following the Wizard of Oz and the Video storyboard activities, full implementations of Voice1-7 and Voice2Q were developed and evaluated in case studies.

A first case study, documented in [51], compared Voice1-7 with the written version of UEQ. The evaluation of 40 participants showed no significant difference between the two implementations, providing initial validity evidence for the direct scale mapping approach.

A second case study evaluated Voice2Q with 40 Computer Science students, ages ranging from 20 to 33, with a median of 21. Of all participants, 80% were male, 15% female, and 5% indicated other or undisclosed.

These participants evaluated two products, alternating the voice-enabled version Voice2Q with the traditional written version, which we call UEQ-W. A Wilcoxon rank test for independent samples was conducted between UEQ-W and Voice2Q data, finding no significant differences in the responses of the six scales of UEQ. Since no p-value was significant, it was not considered necessary to apply a correction for multiple response variables. The information of the means and variances for each implementation scale can be seen in table 1, as well as the Wilcoxon test p-value, which shows that there is no significant difference between both implementations.

Table 1: Means, Variances, and Wilcoxon rank test for independent samples of UEQ-W and Voice2Q.

UEQ Scale	UEQ-W Mean (Var)	Voice2Q Mean (Var)	Wilcoxon p-value
Attractiveness	1.38 (1.26)	1.38 (0.60)	0.7650
Perspicuity	1.31 (1.44)	1.24 (0.69)	0.5174
Efficiency	0.84 (1.12)	0.61 (0.59)	0.2174
Dependability	0.81 (1.20)	0.66 (0.57)	0.4959
Stimulation	1.32 (1.68)	1.04 (0.70)	0.1088
Novelty	1.71 (1.39)	1.59 (0.77)	0.3241

In order to assess the reliability of the Voice2Q instrument per voice, Cronbach's Alpha coefficients of the six UEQ scales were calculated for Voice2Q, based on the correlations between the items of each scale. It is worth noting that the UEQ questionnaire has 26 questions: 6 questions for the Attractiveness scale, and 4 for the 5 scales: Perspicuity, Efficiency, Dependability, Stimulation, and Novelty. For each of these six scales, UEQ calculates Cronbach's Alpha Coefficient as a measure of scale consistency.

Table 2: Comparison of the Cronbach's Alpha coefficients for the Voice2Q version with the test cases presented in the founding paper of UEQ in Spanish.

UEQ Scale	Case 1 UEQ Spanish	Case 2 UEQ Spanish	Voice2Q
Attractiveness	0.85	0.83	0.82
Perspicuity	0.59	0.71	0.73
Efficiency	0.74	0.72	0.63
Dependability	0.48	0.55	0.45
Stimulation	0.75	0.78	0.77
Novelty	0.64	0.71	0.87

The values of the Cronbach's Alpha coefficients for Voice2Q are within the expected ranges, if compared with the coefficients presented by the article that describes the official version of UEQ in Spanish [52], as shown in table 2. In the cited article, two case studies are presented to validate the UEQ translation to Spanish: a first case in which 94 participants evaluated the UX of amazon.com, and a second case in which 95 participants evaluated the Skype program. Authors affirm that these coefficients are appropriate to demonstrate the reliability of the instrument, indicating that the scales are sufficiently consistent. As can be seen, the coefficients obtained with the Voice2Q instrument are similar.

6.4. UX, Usability, and Workload Evaluation

The Voice2Q instrument was evaluated with three different instruments: AttrakDiff questionnaire [6] to measure general UX, SUS [54] for Usability, and NASA-TLX [55] for the specific workload. This evaluation was carried out with 30 participants (different from the 40 participants of the validity evaluation), Computer Science students, aged between 20 and 34 years, with a mean of 22.23 years, 23 male (76.7%) and 7 female (23.3%). To measure general UX, these 30 participants compared the use of the written questionnaire UEQ-W and the voice questionnaire Voice2Q, using the standardized AttrakDiff questionnaire.

The AttrakDiff questionnaire evaluates four components. Pragmatic Quality (PQ) measures whether a product is predictable, confusing, complicated, among others, Hedonic Stimulation (HQ-S) evaluates feelings associated with the product, such as whether it is perceived as boring, interesting, novel or disappointing. Hedonic Identification (HQ-I) measures the ability of a product to connect us with other people instead of isolating them. And Attractiveness component (ATT) represents the overall value of the product based on the perception of pragmatic and hedonic qualities.

As can be seen in figure 7, UEQ-W showed an advantage in PQ, while Voice2Q received the highest rating in HQ-S. Values for HQ-I and ATT were similar across both implementations.

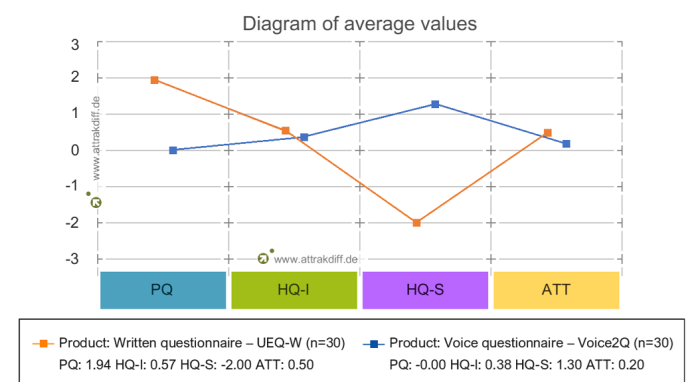


Figure 7: Comparison of the four AttrakDiff scales for UEQ-W (written) and Voice2Q (voice). PQ, HQ-I, HQ-S, ATT stand for Pragmatic Quality, Identification of Hedonic Quality, Stimulation of Hedonic Quality, and Attractiveness, respectively. Chart generated automatically using the official AttrakDiff Data Analysis Tool [6].

Additionally, AttrakDiff classifies the analyzed products in a matrix whose vertical axis depicts the Hedonic quality (considering both Identification and Stimulation) while the horizontal axis represents the Pragmatic quality. Depending on the dimensional values, the product will be in one or more regions. Figure 8 shows that AttrakDiff classifies Voice2Q questionnaire as neutral with a tendency towards the self-oriented region, while the written UEQ-W questionnaire is task-oriented.

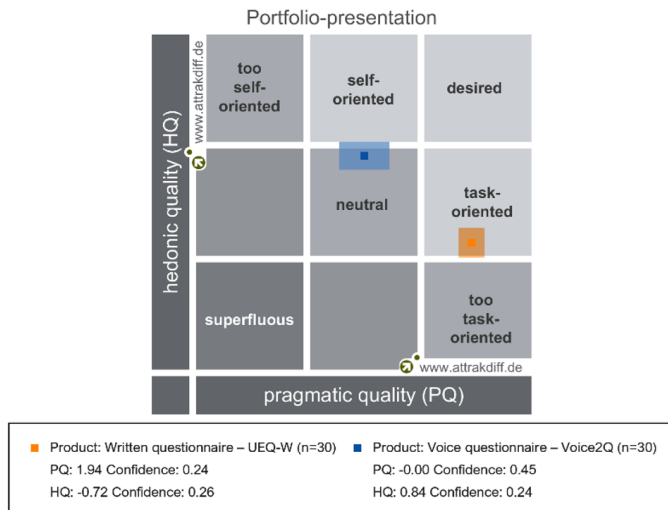


Figure 8: Comparison of Pragmatic and Hedonic Qualities for UEQ-W (written) and Voice2Q (voice). The UEQ-W is classified as task-oriented and the Voice2Q is classified as neutral. Chart generated automatically using the official AttrakDiff Data Analysis Tool [6].

Regarding the specific mental workload, the 30 participants filled the NASA-TLX standardized questionnaire [55], composed of six questions presented as 21-point Likert scales, in which meaning is only shown at the two extremes. Based on these questions, NASA-TLX calculates a general value between 0 and 100, which is established as a benchmark for the workload of the task. Low values correspond to low mental and physical demands, little frustration and success when performing the task.

The overall NASA-TLX score for Voice2Q is 29.7, calculated independently to enable comparisons with other systems that perform similar tasks. For comparison, an evaluation of the written UEQ-W was carried out, with 26 participants (Computer Science students, ages ranging from 19 to 28, with a median of 21 years, 85% male and 15% female) different from the participants of the two previous evaluations. NASA-TLX overall value was 25.3, lower than the 29.7 result for Voice2Q. Additionally, the average time required to complete Voice2Q was 9:44 minutes, while average time to complete UEQ-W was 2:21 minutes.

It is clear that it requires a considerably longer completion time when Voice2Q is used. However, this could be due to the learning curve of using such interfaces. Furthermore, the improvement in accessibility and overall User Experience might balance the completion time, especially if it decreases over time.

As for the Usability evaluation, the System Usability Scale (SUS) was used. This questionnaire has ten items in the form of a five-point Likert scale, to which an arithmetic formula is applied to generate a value between 0 and 100

that represents the global value of system Usability. Participants respond by marking one of five values on a scale, in which only the extreme values of Strongly disagree and Strongly agree are presented. Five of these statements are written in positive terms and five in negative terms, to avoid straight-lining (the tendency to mark answers down on-line without reasoning each question individually), which could occur in some participants if all the questions were written with the same polarity.

A Usability evaluation of Voice2Q was carried out with the 30 participants, where the overall SUS result obtained is 69.7, which is in the lower part of the "Good" category, if compared with the interpretation guide for SUS values [54]. As a comparison, in an evaluation conducted with the 26 participants previously mentioned, the written questionnaire UEQ-W obtained an overall SUS rating of 77.2, which is also located in the "Good" category.

Regarding the validity and UX evaluation of Voice2Q, two interesting points to highlight are the inconsistencies that appear in Voice2Q compared to the written version UEQ-W, and the dispersion of UEQ scales mean responses. Inconsistencies are identified at scale level for each participant. The heuristic applied is as follows: responses to items within scale should not differ by more than three points between the highest value and the lowest value of the 7-point scale. If a participant presents inconsistencies in three or more UEQ scales, their answer is considered critical, and UEQ suggests removing it from the evaluation.

It can be seen in table 3 that Voice2Q presents significantly fewer inconsistencies than the written version UEQ-W, both at the total level (responses in the six scales for the 40 participants represent 240 possible inconsistencies) and at the level of responses considered critical.

Table 3: Inconsistencies identified in UEQ-W and Voice2Q.

Format	Inconsistencies	Critical answers
UEQ-W	37/240 (15.4 %)	5/40 (12.5 %)
Voice2Q	17/240 (7.1 %)	0/40 (0.0 %)

As for the mean responses of the UEQ scales, the Voice2Q means appear with a smaller dispersion than those presented in UEQ-W: the interquartile ranges are smaller and the minimum and maximum values of these averages are not as far apart, as shown in figure 9.

6.5. Evaluation of Voice2Q+VC

To address challenges identified in the voice-only implementation —particularly mental workload and extended completion time— a new variant was developed: Voice2Q+VC. This enhanced version incorporates minimal visual cues while maintaining voice as the sole input method, without compromising the voice-first interaction paradigm.

A study with 20 participants evaluated Voice2Q+VC across three dimensions: general UX using the AttrakDiff questionnaire, workload using NASA-TLX, and Usability using SUS.

As can be seen in table 4, workload assessments revealed a NASA-TLX score of 27.2 for Voice2Q+VC, lower than the 29.7 for Voice2Q and approaching the 25.3 score

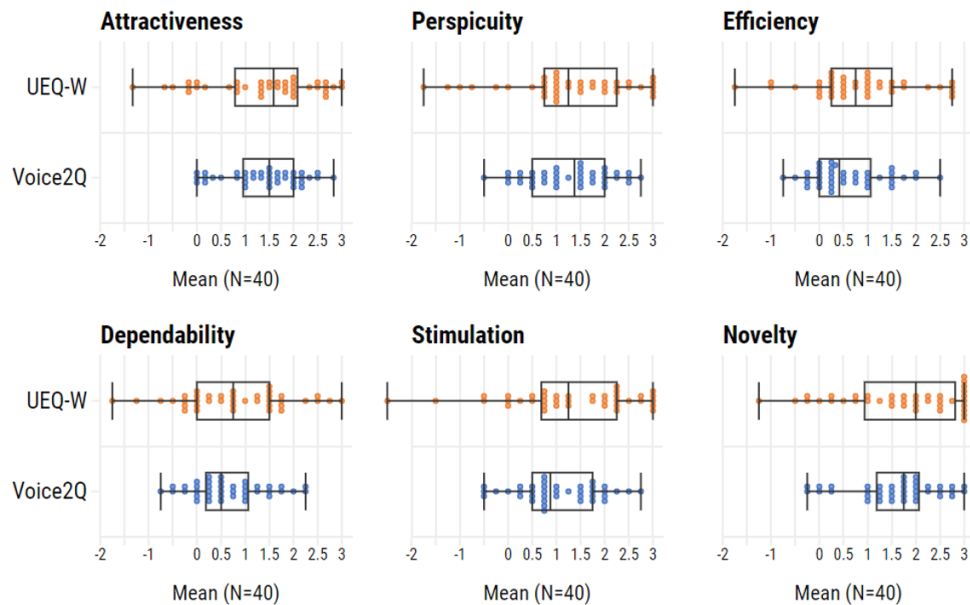


Figure 9: Compared scale dispersion of mean responses for UEQ-W and Voice2Q.

of UEQ-W. This reduction in workload was particularly notable in mental demand and frustration. Usability, measured by SUS, rated Voice2Q+VC at 78.1, higher than 69.7 for Voice2Q and slightly surpassing UEQ-W's score of 77.2. Additionally, the time required to complete Voice2Q+VC averaged 6:42 minutes, shorter than Voice2Q's 9:44 minutes, though still longer than UEQ-W's 2:21 minutes.

As for the UX evaluation, figure 10 presents the comparison for the four AttrakDiff scales. Results show that in general UX, Voice2Q+VC outperformed the traditional written UEQ-W in Hedonic quality metrics, particularly Hedonic stimulation, while showing slight disadvantage in the Pragmatic quality.

Table 4: Usability, Workload, and completion time for the three implementations of UEQ: written, voice only and voice with visual cues.

Questionnaire Implementation	SUS Score	NASA-TLX Score	Average Completion Time
UEQ-W	77.2	25.3	2.21 min
Voice-2Q	69.7	29.7	9.44 min
Voice-2Q+VC	78.1	27.2	6:42 min

Additionally, figure 11 shows a comparison between the AttrakDiff Portfolio-presentation between Voice2Q (in blue) and Voice2Q+VC (in green), where the implementation with visual cues moved closer to the "desired" quadrant, indicating an improvement in overall UX.

These findings suggest that the addition of visual cues in Voice2Q+VC effectively mitigates some challenges of voice-only interfaces while preserving the natural interaction advantages inherent to voice-first systems.

As a final note, table 5 shows the number of participants covered in this study, for design and evaluation phases. Table 5 includes the 40 participants who validated the direct Voice1-7 implementation, work described in detail in [51].

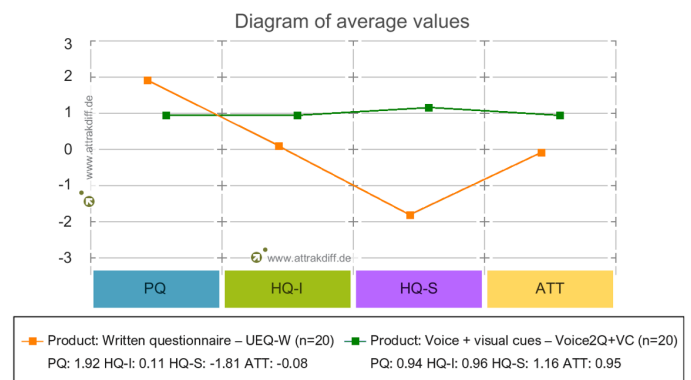


Figure 10: Comparison of the four AttrakDiff scales for UEQ-W (written) and Voice2Q+VC (voice + visual cues). PQ, HQ-I, HQ-S, ATT stand for Pragmatic Quality, Identification of Hedonic Quality, Stimulation of Hedonic Quality, and Attractiveness, respectively. Chart generated automatically using the official AttrakDiff Data Analysis Tool [6].

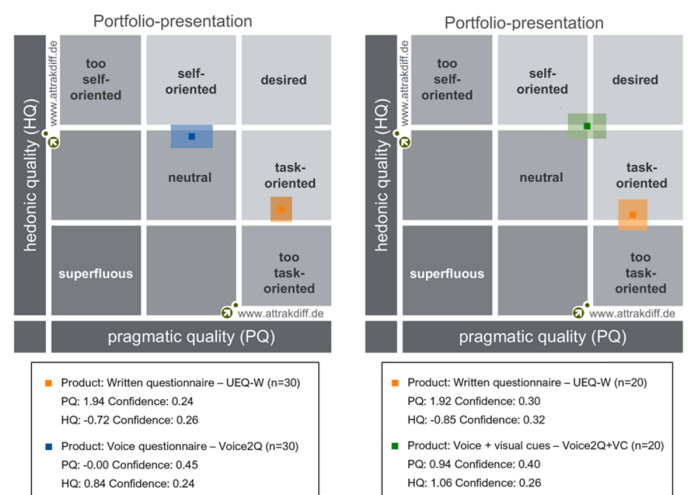


Figure 11: Comparison of Voice2Q and Voice2Q+VC AttrakDiff quality matrix. Voice2Q+VC is almost considered desired or self-oriented. Chart created from two charts generated by AttrakDiff's official data analysis tool [6].

Table 5: Number of research participants, by phase and activity.

Phase	Activity /Implementation	Number of participants
Design	Wizard of Oz/Voice2Q	12
	VideoStoryboard/Voice2Q	197
	Validity/Voice1-7	40
Evaluation	Validity/Voice2Q	40
	UX, Usability/Voice2Q	30
	UX, Usability/Voice2Q+VC	26
Total		345

7. Conclusions

The results of this research show that a version of the UEQ questionnaire, whose capture mechanism is done by voice, provides results without significant differences from an evaluation of the same product using the traditional written UEQ. In addition, the results are appropriate in terms of the reliability of the questionnaire, comparing Cronbach's Alpha coefficients with the coefficients presented in the founding paper of UEQ in Spanish [52].

In terms of overall UX, the traditional written UEQ-W outmatches Voice2Q in AttrakDiff Pragmatic quality, while Voice2Q performs significantly better in Stimulation quality. The qualities of identification and attractiveness would be equivalent. Therefore, UEQ-W would be more task-oriented, while Voice2Q would be more self-oriented, in terms of the theoretical model of UX presented in [56]. The version with visual cues (Voice2Q+VC) outperformed written UEQ-W in both Hedonic quality metrics, stimulation and identification, and also in Attractiveness, while slightly underperforming in the Pragmatic quality. Voice2Q+VC would also perform better in the portfolio-representation of the AttrakDiff evaluation, balancing the self-oriented and task-oriented classifications, almost reaching the desired quadrant.

Regarding the complementary evaluation of Usability using SUS scores, all three questionnaires correspond to the "Good" range of the general interpretation guide of SUS, Voice2Q+VC being the questionnaire with the highest score (78.1), followed by UEQ-W (77.2), and finally Voice2Q (69.7). In terms of specific workload, the traditional written UEQ-W outperforms both voice implementations, although Voice2Q+VC comes close to overall NASA-TLX score for UEQ-W (27.2 versus 25.3). A similar performance is seen in terms of average time to complete the questionnaire, with UEQ-W having the shortest time compared to the two voice implementations.

Although voice implementations require more time for the participant to complete, it is important to mention that the responses show significantly fewer inconsistencies than those obtained with the written UEQ-W questionnaire, and the means of the responses obtained with Voice2Q show smaller dispersion than those obtained with UEQ-W.

In summary, the results obtained with the Voice2Q implementation are promising, in the sense that voice could be used as an alternative method for capturing responses for UX evaluations, with results similar to those obtained with the written version of UEQ and, at the same time, with fewer inconsistencies in participants' responses.

This research demonstrates that voice-based implementations of UX questionnaires, such as Voice2Q, are valid

and reliable alternatives to traditional written formats. The inclusion of conversational patterns and branching questioning ensures accurate and engaging user interactions, reducing inconsistencies and reducing response dispersion. While Voice2Q offers significant advantages in Hedonic qualities like stimulation and identification, its limitations in Pragmatic efficiency highlight opportunities for enhancement.

The development of Voice2Q+VC, incorporating minimal visual cues, addresses some of the Voice2Q (voice-only) limitations by lowering mental workload and reducing the completion time, while maintaining the core benefits of voice implementation. Compared to Voice2Q, the Voice2Q+VC variant moves closer to the ideal balance of pragmatic and hedonic qualities, as evidenced by its improved scores in AttrakDiff, NASA-TLX, and SUS evaluations.

7.1. Limitations

It is important to mention that generalization is not the goal of this research. Due to the sampling of participants and experiments conducted, we acknowledge that this exploration provides initial empirical evidence supporting different ways to collect UX evaluation responses using voice interfaces.

Furthermore, this study did not aim to evaluate the effectiveness of voice interfaces in a broad sense. As such, we recognize that further evaluations considering the role of voice inputs and outputs (e.g., volume, pitch and clarity) could further be considered. While this limitation may affect the experimental experiences of the study, it does not compromise the internal validity or the integrity of our findings.

Finally, even though it could be considered a limitation, the number of people that participated in this study is consistent with the literature. The participation of sample sizes for standardized questionnaires is consistent according to [7] and for NASA TLX according to [57].

7.2. Future work

As future work, several lines of research are identified from this work. Firstly, it is possible to extend the research on voice implementations of UX evaluation questionnaires with the incorporation of other identifiable patterns. For example, variant described in [58] implements the semantic differential as branching questions. In this approach, when a participant selects the central or neutral option, a follow-up question is posed to determine whether there is any slight tendency of the attitude towards one side of the scale. The three central positions of the semantic differential would be represented by the central option and its adjacent positions. If the participant initially selects a direction, the follow-up question offers two intensity options, for example very much and extremely, corresponding to the two outer boxes on each side of the scale. This implementation, different from the one used in this research work, would emphasize identifying whether the branching questions are prone to a central tendency of the responses.

Another variant to consider is the voice implementation of the official short version of the UEQ [14], which has eight

items compared to the usual 26-item questionnaire. These eight items are only presented in three scales: a Pragmatic scale averaging four of the items, a Hedonic scale, with the other four items, and a Global scale with the average of the eight items. The creators of UEQ recommend using this questionnaire only in situations where a full UEQ cannot be administered, mainly due to issues related to the time it takes to complete the questionnaire, since it does not allow for detailed measurement of the six subscales present in the original UEQ. The short version UEQ-S only allows for approximate measurement in higher-level meta-dimensions, which provides less detail than the information provided for the six subscales. This is consistent with what was stated in [59] on the ability of the traditional UEQ to measure the three most important aspects of the UX model proposed in [56]: Pragmatic quality, Hedonic quality, and General attractiveness.

In this same line of work, a version of the complete UEQ could be implemented, but as the user's responses are collected, it would be calculated whether the values of the scale items are consistent for the scale (i.e., they do not generate inconsistencies). If after collecting, for example, three of the four items of a scale, the responses are strongly related, the fourth item of that scale could be omitted, dynamically reducing the length of the questionnaire.

Another evaluation that could be carried out is a study with participants using the voice-implemented questionnaire more than once at different times, so that the impact of familiarity with the instrument, the time to fill out the questionnaire and the specific workload can then be measured. Until now, the evaluations have been cross-sectional, where participants are confronted with the voice questionnaire for the first time, compared to the written version of the questionnaire, which shares many formatting elements with other questionnaires that participants have used before and to which they are already accustomed.

Another line of research would be the identification of conversational patterns that convert to voice other question formats, for example Likert scales, which are present in standardized questionnaires such as the meCUE, the SUS questionnaire, and the NASA-TLX. A study of this type would help identify the challenges of this transformation and eventually also propose voice variants for these questionnaires.

Finally, it is considered pertinent to explore the adaptation of some of the elements used in the voice implementations of this research work to questionnaires with visual representation, for example through a Web page. For example, branching questions could be used instead of presenting the semantic differentials of the UEQ in their traditional form, so that the direction of the attitude is separated from the intensity of the attitude. These branching questions would appear dynamically as the questionnaire is filled out. In addition, the pattern in which the concepts being asked are explained could also be implemented. The questionnaire could be designed so that, if the participant hovers the mouse pointer over the concepts being asked, a dialog box is displayed with the description of the concept in the context of the UX evaluation, thus helping reduce ambiguities and obtain higher quality responses.

Data Availability The datasets generated during the current study are not publicly available but can be provided from the corresponding author on request.

Threats to Validity As with any empirical study, our conclusions are context dependent. We acknowledge that increasing the number of participants in the presented case studies would reduce variance and enhance confidence in the results.

Conflict of Interest The authors declare no conflicts of interest.

Acknowledgments The authors express their deep gratitude to Dr. Luis A. Guerrero for his fundamental role in the research that led to this article. His guidance and intellectual contributions were essential to the development of the results presented here.

The development of this manuscript was partially supported by ECCI and CITIC at the Universidad de Costa Rica, under Grant No. 834-C1-013.

References

- [1] J. A. Krosnick, M. K. Berent, "Comparisons of party identification and policy preferences: The impact of survey question format", *American Journal of Political Science*, pp. 941–964, 1993, doi:10.2307/2111580.
- [2] E. Cho, M. D. Molina, J. Wang, "The effects of modality, device, and task differences on perceived human likeness of voice-activated virtual assistants", *Cyberpsychology, Behavior, and Social Networking*, vol. 22, no. 8, pp. 515–520, 2019, doi:10.1089/cyber.2018.0571.
- [3] J. K. Höhne, K. Gavras, J. Claassen, "Typing or speaking? comparing text and voice answers to open questions on sensitive topics in smartphone surveys", *Social Science Computer Review*, vol. 42, no. 4, pp. 1066–1085, 2024, doi:10.1177/08944393231160961.
- [4] B. Laugwitz, T. Held, M. Schrepp, "Construction and evaluation of a user experience questionnaire", "Symposium of the Austrian HCI and usability engineering group", pp. 63–76, Springer, 2008, doi:10.1007/978-3-540-89350-9_6.
- [5] C. Lallemand, G. Gronier, *Méthodes de design UX: 30 méthodes fondamentales pour concevoir et évaluer les systèmes interactifs*, Editions Eyrolles, 2015.
- [6] M. Hassenzahl, M. Burmester, F. Koller, "Attrakdiff: Ein fragebogen zur messung wahrgenommener hedonischer und pragmatischer qualität", "Mensch & computer 2003: interaktion in bewegung", pp. 187–196, Springer, 2003, doi:10.1007/978-3-322-80058-9_19.
- [7] I. Díaz-Oreiro, G. López, L. Quesada, L. A. Guerrero, "Ux evaluation with standardized questionnaires in ubiquitous computing and ambient intelligence: a systematic literature review", *Advances in Human-Computer Interaction*, vol. 2021, no. 1, p. 5518722, 2021, doi:10.1155/2021/5518722.
- [8] D. Norman, J. Miller, A. Henderson, "What you see, some of what's in the future, and how we go about doing it: Hi at apple computer", "Conference companion on Human factors in computing systems", p. 155, 1995, doi:10.1145/223355.223477.
- [9] E. Law, V. Roto, A. P. Vermeeren, J. Kort, M. Hassenzahl, "Towards a shared definition of user experience", "CHI '08 Extended Abstracts on Human Factors in Computing Systems", CHI EA '08, p. 2395–2398, Association for Computing Machinery, New York, NY, USA, 2008, doi:10.1145/1358628.1358693.
- [10] V. Roto, E.-C. Law, A. Vermeeren, J. Hoonhout, *User experience white paper: Bringing clarity to the concept of user experience*, s.n., 2011, geen ISBN Result from Dagstuhl seminar on demarcating user experience, september 15-18, 2010.

- [11] C. Lallemand, G. Gronier, V. Koenig, "User experience: A concept without consensus? exploring practitioners' perspectives through an international survey", *Computers in human behavior*, vol. 43, pp. 35–48, 2015, doi:[10.1016/j.chb.2014.10.048](https://doi.org/10.1016/j.chb.2014.10.048).
- [12] I. DIS, "9241-210: 2010. ergonomics of human system interaction-part 210: Human-centred design for interactive systems", *International Standardization Organization (ISO). Switzerland*, vol. 2, 2009.
- [13] A. P. Vermeeren, E. L.-C. Law, V. Roto, M. Obrist, J. Hoonhout, K. Väänänen-Vainio-Mattila, "User experience evaluation methods: current state and development needs", "Proceedings of the 6th Nordic conference on human-computer interaction: Extending boundaries", pp. 521–530, 2010, doi:[10.1145/1868914.1868973](https://doi.org/10.1145/1868914.1868973).
- [14] A. Hinderks, "Design and evaluation of a short version of the user experience questionnaire (ueq-s)", *International Journal of Interactive Multimedia and Artificial Intelligence*, 2017, doi:[10.9781/ijimai.2017.09.001](https://doi.org/10.9781/ijimai.2017.09.001).
- [15] D. Wallach, J. Conrad, T. Steimle, "The ux metrics table: A missing artifact", "International conference of design, user experience, and usability", pp. 507–517, Springer, 2017, doi:[10.1007/978-3-319-58634-2_37](https://doi.org/10.1007/978-3-319-58634-2_37).
- [16] G. Gronier, C. Lallemand, A. Chauvet, "Mesurer la formation de la première impression d'une interface à l'aide du test des 5 secondes", "Huitieme Colloque de Psychologie Ergonomique (EPIQUE)", 2015.
- [17] V. Roto, M. Obrist, K. Väänänen-Vainio-Mattila, "User experience evaluation methods in academic and industrial contexts", "Proceedings of the Workshop UXEM", vol. 9, pp. 1–5, 2009.
- [18] C. Lallemand, V. Koenig, "How could an intranet be like a friend to me? why standardized ux scales don't always fit", "Proceedings of the European Conference on Cognitive Ergonomics", pp. 9–16, 2017, doi:[10.1145/3121283.3121288](https://doi.org/10.1145/3121283.3121288).
- [19] C. L. B. Maia, E. S. Furtado, "A systematic review about user experience evaluation", "International conference of design, user experience, and usability", pp. 445–455, Springer, 2016, doi:[10.1007/978-3-319-40409-7_42](https://doi.org/10.1007/978-3-319-40409-7_42).
- [20] Y. Forster, S. Hergeth, F. Naujoks, J. F. Krems, "How usability can save the day-methodological considerations for making automated driving a success story", "Proceedings of the 10th international conference on automotive user interfaces and interactive vehicular applications", pp. 278–290, 2018, doi:[10.1145/3239060.3239076](https://doi.org/10.1145/3239060.3239076).
- [21] J. Klammer, F. W. van den Anker, "A platform to connect swiss consumers of fair trade products with producers in developing countries: needs and motivations", "International Conference of Design, User Experience, and Usability", pp. 664–681, Springer, 2018, doi:[10.1007/978-3-319-91806-8_52](https://doi.org/10.1007/978-3-319-91806-8_52).
- [22] J. Baumgartner, A. Sonderegger, J. Sauer, "No need to read: Developing a pictorial single-item scale for measuring perceived usability", *International Journal of Human-Computer Studies*, vol. 122, pp. 78–89, 2019, doi:[10.1016/j.ijhcs.2018.08.008](https://doi.org/10.1016/j.ijhcs.2018.08.008).
- [23] D. Wigdor, D. Wixon, *Brave NUI world: designing natural user interfaces for touch and gesture*, Elsevier, 2011.
- [24] T. Nishida, *Conversational informatics: An engineering approach*, John Wiley & Sons, 2008.
- [25] D. Frohlich, P. Luff, "Applying the technology of conversation to the technology for conversation", "Computers and conversation", pp. 187–220, Elsevier, 1990.
- [26] V. K. Chaudhri, A. Cheyer, R. Guili, W. Jarrold, K. L. Myers, J. Niekarsz, "A case study in engineering a knowledge base for an intelligent personal assistant.", "SemDesk", pp. 25–32, 2006.
- [27] D. M. Kaushik, R. Jain, "Natural user interfaces: Trend in virtual interaction", *arXiv preprint arXiv:1405.0101*, 2014.
- [28] E. A. Schegloff, *Sequence organization in interaction: A primer in conversation analysis I*, vol. 1, Cambridge university press, 2007, doi:[10.1017/CBO9780511791208](https://doi.org/10.1017/CBO9780511791208).
- [29] D. A. Norman, *Living with complexity*, MIT press, 2016.
- [30] R. J. Moore, R. Arar, "Conversational ux design: an introduction", "Studies in conversational UX design", pp. 1–16, Springer, 2018, doi:[10.1007/978-3-319-95579-7_1](https://doi.org/10.1007/978-3-319-95579-7_1).
- [31] R. J. Moore, R. Arar, *Conversational UX design: A practitioner's guide to the natural conversation framework*, Morgan & Claypool, 2019, doi:[10.1145/3304087](https://doi.org/10.1145/3304087).
- [32] H. Sacks, E. A. Schegloff, G. Jefferson, "A simplest systematics for the organization of turn-taking for conversation", *Language*, vol. 50, no. 4, pp. 696–735, 1974, doi:[10.1353/lan.1974.0010](https://doi.org/10.1353/lan.1974.0010).
- [33] A. B. Kocaballi, L. Laranjo, E. Coiera, "Understanding and measuring user experience in conversational interfaces", *Interacting with Computers*, vol. 31, no. 2, pp. 192–207, 2019, doi:[10.1093/iwc/iwz022](https://doi.org/10.1093/iwc/iwz022).
- [34] A. L. Iniguez-Carrillo, L. S. Gaytan-Lugo, M. A. Garcia-Ruiz, R. Maciel-Arellano, "Usability questionnaires to evaluate voice user interfaces", *IEEE Latin America Transactions*, vol. 19, no. 9, pp. 1468–1477, 2021, doi:[10.1109/TLA.2021.9477283](https://doi.org/10.1109/TLA.2021.9477283).
- [35] J. R. Lewis, "Standardized questionnaires for voice interaction design", *Voice Interaction Design*, vol. 1, no. 1, pp. 1–16, 2016.
- [36] F. Iniesto, T. Coughlan, K. Lister, "Implementing an accessible conversational user interface: applying feedback from university students and disability support advisors", "Proceedings of the 18th International Web for All Conference", pp. 1–5, 2021, doi:[10.1145/3430263.3452436](https://doi.org/10.1145/3430263.3452436).
- [37] J. Wei, W. Jiang, C. Wang, D. Yu, J. Goncalves, T. Dingler, V. Kostakos, "Understanding how to administer voice surveys through smart speakers", *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, no. CSCW2, pp. 1–32, 2022, doi:[10.1145/3555767](https://doi.org/10.1145/3555767).
- [38] A. Barbaric, C. Munteanu, H. Ross, J. A. Cafazzo, "A voice app design for heart failure self-management: proof-of-concept implementation study", *JMIR Formative Research*, vol. 6, no. 12, p. e40021, 2022, doi:[10.2196/40021](https://doi.org/10.2196/40021).
- [39] J.-G. Shin, G.-Y. Choi, H.-J. Hwang, S.-H. Kim, "Evaluation of emotional satisfaction using questionnaires in voice-based human-ai interaction", *Applied Sciences*, vol. 11, no. 4, p. 1920, 2021, doi:[10.3390/app11041920](https://doi.org/10.3390/app11041920).
- [40] E. Kuang, E. Jahangirzadeh Soure, M. Fan, J. Zhao, K. Shinohara, "Collaboration with conversational ai assistants for ux evaluation: Questions and how to ask them (voice vs. text)", "Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems", pp. 1–15, 2023, doi:[10.1145/3544548.3581247](https://doi.org/10.1145/3544548.3581247).
- [41] B. Zarouali, T. Araujo, J. Ohme, C. De Vreese, "Comparing chatbots and online surveys for (longitudinal) data collection: an investigation of response characteristics, data quality, and user evaluation", *Communication Methods and Measures*, vol. 18, no. 1, pp. 72–91, 2024, doi:[10.1080/19312458.2023.2210576](https://doi.org/10.1080/19312458.2023.2210576).
- [42] E. A. Beam, "Social media as a recruitment and data collection tool: Experimental evidence on the relative effectiveness of web surveys and chatbots", *Journal of Development Economics*, vol. 162, p. 103069, 2023, doi:[10.1016/j.jdeveco.2023.103069](https://doi.org/10.1016/j.jdeveco.2023.103069).
- [43] H. Soni, J. Ivanova, H. Wilczewski, A. Bailey, T. Ong, A. Narma, B. E. Bunnell, B. M. Welch, "Virtual conversational agents versus online forms: patient experience and preferences for health data collection", *Frontiers in Digital Health*, vol. 4, p. 954069, 2022, doi:[10.3389/fdgth.2022.954069](https://doi.org/10.3389/fdgth.2022.954069).
- [44] P. Sprengholz, C. Betsch, "Ok google: Using virtual assistants for data collection in psychological and behavioral research", *Behavior Research Methods*, vol. 54, no. 3, pp. 1227–1239, 2022, doi:[10.3758/s13428-021-01629-y](https://doi.org/10.3758/s13428-021-01629-y).
- [45] I. Celino, G. R. Calegari, "Submitting surveys via a conversational interface: An evaluation of user acceptance and approach effectiveness", *International Journal of Human-Computer Studies*, vol. 139, p. 102410, 2020, doi:[10.1016/j.ijhcs.2020.102410](https://doi.org/10.1016/j.ijhcs.2020.102410).
- [46] R. Maharjan, D. A. Rohani, P. Bækgaard, J. Bardram, K. Doherty, "Can we talk? design implications for the questionnaire-driven self-report of health and wellbeing via conversational agent", "Proceedings of the 3rd Conference on Conversational User Interfaces", pp. 1–11, 2021, doi:[10.1145/3469595.3469600](https://doi.org/10.1145/3469595.3469600).

- [47] P. V. Miller, "Alternative question forms for attitude scale questions in telephone interviews", *Public Opinion Quarterly*, vol. 48, no. 4, pp. 766–778, 1984, doi:[10.1086/268879](https://doi.org/10.1086/268879).
- [48] L. R. Fabrigar, J. A. Krosnick, "Attitude measurement and questionnaire design", *Blackwell encyclopedia of social psychology*, pp. 42–47, 1995.
- [49] J. H. Yu, G. Albaum, M. Swenson, "Is a central tendency error inherent in the use of semantic differential scales in different cultures?", *International Journal of Market Research*, vol. 45, no. 2, pp. 1–16, 2003, doi:[10.1177/147078530304500201](https://doi.org/10.1177/147078530304500201).
- [50] V. D. de Rada Igúzquiza, "¿influye el diseño de las preguntas en las respuestas de los entrevistados?", *Revista Española de Sociología*, vol. 31, no. 1, p. a83, 2022, doi:[10.22325/fes/res.2022.83](https://doi.org/10.22325/fes/res.2022.83).
- [51] J. C. Mata-Serrano, I. Díaz-Oreiro, G. López, L. A. Guerrero, "Comparing written and voice captured responses of the user experience questionnaire (ueq)", "International Conference on Information Technology & Systems", pp. 519–529, Springer, 2022, doi:[10.1007/978-3-030-96293-7_43](https://doi.org/10.1007/978-3-030-96293-7_43).
- [52] M. Rauschenberger, M. Schrepp, M. Pérez Cota, S. Olschner, J. Thomaschewski, "Efficient measurement of the user experience of interactive products. how to use the user experience questionnaire (ueq). example: Spanish language version", *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 2, no. 1, pp. 39–45, 2013, doi:[10.9781/ijimai.2013.215](https://doi.org/10.9781/ijimai.2013.215).
- [53] I. Díaz-Oreiro, G. López, L. Quesada, L. A. Guerrero, "Conversational design patterns for a ux evaluation instrument implemented by voice", "International Conference on Information Technology & Systems", pp. 530–540, Springer, 2022, doi:[10.1007/978-3-030-96293-7_44](https://doi.org/10.1007/978-3-030-96293-7_44).
- [54] J. Brooke, *et al.*, "Sus-a quick and dirty usability scale", *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996, doi:[10.1201/9781498710411-35](https://doi.org/10.1201/9781498710411-35).
- [55] S. G. Hart, L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research", "Advances in psychology", vol. 52, pp. 139–183, Elsevier, 1988, doi:[10.1016/S0166-4115\(08\)62386-9](https://doi.org/10.1016/S0166-4115(08)62386-9).
- [56] M. Hassenzahl, "The effect of perceived hedonic quality on product appealingness", *International Journal of Human-Computer Interaction*, vol. 13, no. 4, pp. 481–499, 2001, doi:[10.1207/S15327590IJHC1304_07](https://doi.org/10.1207/S15327590IJHC1304_07).
- [57] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later", *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, pp. 904–908, 2006, doi:[10.1177/154193120605000909](https://doi.org/10.1177/154193120605000909).
- [58] R. Wang, J. A. Krosnick, "Middle alternatives and measurement validity: A recommendation for survey researchers", *International Journal of Social Research Methodology*, vol. 23, no. 2, pp. 169–184, 2020, doi:[10.1080/13645579.2019.1645384](https://doi.org/10.1080/13645579.2019.1645384).
- [59] A. Schankin, M. Budde, T. Riedel, M. Beigl, "Psychometric properties of the user experience questionnaire (ueq)", "Proceedings of the 2022 Chi conference on human factors in computing systems", pp. 1–11, 2022, doi:[10.1145/3491102.3502098](https://doi.org/10.1145/3491102.3502098).

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



Ignacio Díaz-Oreiro has over 30 years in software engineering, with over two decades in industry and the last eight years devoted to teaching and research in Computer Science. He earned his PhD in Computer Science in 2025.

His interests include User Experience, usability and accessibility, software quality, and software testing. Throughout his academic career, he has authored 20 peer-reviewed publications, including journal articles, conference papers, and book chapters.



Gustavo Lopez is a Full Professor and researcher with nearly 15 years of experience in Computer Science. He received his PhD in Computer Science in 2019.

His areas of interest include: Usability, User Experience and accessibility, human-centered design, software process improvement, and the implementation of agile practices. He has published over 100 peer-reviewed research articles, of which more than 20 have been in indexed academic journals.

Superconductivity Research Method Based on Atomic Properties and Hole-Electron Tensor

Guohua Zhang^{*1,2} , Xiyu Zhang³

¹ Shenzhen Polytechnic University, Shenzhen, 518000, China

² R&D Department Shenzhen Hengdrive Technologies Co. Ltd, Shenzhen, 518100, China

³ Israel Institute of Technology Guangdong, Shantou, 515000, China

Email(s): Hansun.zhang@hengdrive.com (Guohua zhang), zhang52016@gtit.edu.cn (xiyu zhang)

*Corresponding author: Guohua Zhang, Shanhai Shangcheng, Tongda Road, Xixiang, Baoan, Shenzhen, Guangdong, 518103, China. Email: 617620347@qq.com

ABSTRACT: The microscopic mechanism of superconducting critical temperature (T_c) remains a long-standing unsolved problem in condensed matter physics. Based on de Broglie's matter wave theory, this work introduces the concepts of superconducting gene (SCG) and superconducting DNA (SCD). A phenomenological model is proposed that relates T_c to the number of orbital electron-hole pairs (N), the mass of SCD, and the diameter of SCG. The model is validated across multiple material families, including copper oxides, iron-based superconductors, and high-pressure hydrides.

It is found that the critical temperature of superconductivity can be effectively increased by adjusting the electron orbital filling (N), SCG (a), and introducing light element heterojunctions. Based on this, the "orbital engineering" strategy is proposed, which provides a new path to break through the bottleneck of room-temperature superconductivity. In addition, isotope effect of all superconductors is verified by the general superconducting formula, which further guides the design of effective components of high-temperature superconductors.

This work attempts to incorporate the isotope effect of BCS theory superconductor and strong correlation theory superconductor into a unified phenomenological framework, providing a consistent description of the critical temperature mechanism across different types of superconductors.

A hole-electron model based on electron and hole arrangement is constructed, which can dynamically describe the formation, division and recombination processes of carrier pairs, so as to more accurately simulate the superconductivity mechanism.

This study not only makes important theoretical progress, but also provides a solid foundation and innovative ideas for future design and development of new superconducting materials. By combining pressure/doping regulation, light element heterojunction design and other methods, it is expected to achieve superconducting materials with higher critical temperatures and promote the application and development of superconducting technology.

KEYWORDS: Superconduct, BCS, Isotope effect

1. Introduction: The century-old evolution and unsolved mysteries of SC theory

Since the discovery of superconductivity in mercury in 1911 [1], superconductivity research has progressed through three major milestones.

The BCS theory [2], proposed in 1957, successfully explains conventional superconductors such as Nb_3Sn ($T_c \sim 18$ K) based on Cooper pair formation via electron-phonon coupling, but predicts a theoretical limit of $T_c \leq 40$ K [2].

The discovery of high-temperature superconductivity in the La-Ba-Cu-O system in 1986 [3] revolutionized the field, challenging the BCS paradigm and leading to the development of new theoretical frameworks including strong correlation theory and resonant valence bonding.

More recently, the discovery of superconductivity in H_2S at 200 K under 150 GPa [4] opened a new dimension of high-pressure superconductivity research.

However, the existing theories still face three major challenges:

- For copper oxides, the competition between pseudogap, striped phase, and superconducting state in doping phase diagrams remains unresolved [5];
- For iron-based superconductors, the symmetry of multi-orbital magnetic fluctuations and the nature of superconducting order parameters are still under debate [6];
- For room-temperature superconductivity, the stability and mechanism of carbonaceous sulfur hydrides (e.g., C-S-H, $T_c \sim 287$ K [7]) remain controversial.

In addition to these experimental challenges, existing theoretical approaches also have significant limitations. Eliashberg theory, while successful for conventional superconductors, becomes computationally intractable for strongly correlated systems [8]. Strong correlation models based on the Hubbard Hamiltonian can capture many features of cuprate superconductivity but lack predictive power for T_c [9]. Recent machine learning approaches have shown promise for T_c prediction but provide limited physical insight [10].

This paper aims to construct a bridge and quantitatively correlate microscopic electronic structures with macroscopic TC, providing a new paradigm for room-temperature superconducting design.

2. Theoretical Part

2.1. BCS theory and its extension

At the heart of the BCS theory lies the cooperative cohesion induced by electron-phonon coupling, which has a T_c formula of:

$$T_c = 1.13\omega_D \exp\left(-\frac{1}{N(E_F)V}\right) \quad (1)$$

here ω_D : the Debye freq; $N(E_F)$: the Fermi plane density;
 V : the attraction potential.

- Successful case: Double-band superconductivity of MgB_2 ($T_c=39\text{K}$ [11]) can be accurately described by the Eliashberg equation [8];

- Limitations: It cannot explain the d-wave pairing and pseudo-bandgap phenomena in copper oxides [9].

2.2. Strong correlation with the Hubbard model

For copper oxides, Mott physics and Hubbard models have become mainstream:

- T-J model: hole doping leads to spin singlet resonances in antiferromagnetic background [10];
- Pseudogap origin: preformed Cooper pair vs topological order parameter [12].
- Controversy: ARPES displays Fermi arcs instead of closed surfaces [13], challenging traditional paired images.

2.3. Quantum critical fluctuation mechanism

In iron-based superconductors, spin fluctuations near the antiferromagnetic quantum critical point mediate $s\pm$ -wave pairing [14], but cannot explain pressure-induced *non-monotonic changes in T_c* [15].

2.4. Revival of high-pressure hydride and electron-phonon coupling

The ultra-high T_c of H_2S and LaH_{10} re-highlights the importance of electron-phonon coupling [16], but first-principles calculations show that:

- High-frequency phonons (H-H bond vibration) dominate the coupling strength [17];
- Electron associations in metal hydrides cannot be ignored [18].

2.5. Acoustic excitons excitation model

Phonon can be treated as matter wave, the energy of the acoustic exciton is expressed as $\varepsilon_k = \hbar_k v_0$ ($v_0 = v_1$), v_1 is the frequency of the carrier. This model is a thermal excitation model of monotonic excitons, which is called the mono excitation model [19]. Image:



Figure 1. Acoustic excitons excitation model

2.6. No dissipative conductivity mechanism

Before and after the acoustic excitation process, the crystal lattice and carriers do not change in any way, but there are acoustic excitons in the middle that participate in the conductive process of the carriers. (Figure below)

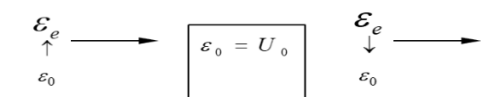


Figure 2. Non-dissipative conductive mechanism [19]

In this state, the conductor will not consume energy to complete the conductive process. In this proposed

mechanism, conduction occurs without net energy dissipation because the energy absorbed from phonons during carrier acceleration is exactly balanced by the energy released during deceleration.

When acoustic excitons participate in conduction as intermediate states, the energy of the initial and final state grids and carriers is conserved, and the phonon absorption and emission process cancel out, and the total energy remains unchanged. Electrons achieve directional motion by periodically exchanging acoustic excitons without overcoming lattice scattering, so the resistance is zero.

2.6.1. The band structure of electrons in a one-dimensional periodic potential

Generally, the conductor of the solid phase is composed of crystals. The movement of carriers in crystals can be simplified to move in 1D periodic field. The 1D periodic field $U(x)$ as shown in the figure consists of a square barrier.

$$U(x) = U(x + n_0 a) \quad (2)$$

here n_0 is an integer; a is the lattice dimension of the 1D lattice; $a=c+b$; and $U(x)$ is periodic [19].

$$U(x) = \begin{cases} 0 & (n_0 a \leq x \leq n_0 a + c) \\ U_0 & (n_0 a - b < x < n_0 a) \end{cases} \quad (3)$$

The image is:

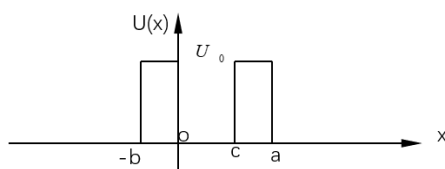


Figure 3. One-dimensional periodic square potential trap [19]

According to the Kronig-Penner model, the electron wavefunction satisfies Bloch's theorem $\psi(x+a)=eika\psi(x)$, and solves the Schrödinger equation in sections.

The wavefunctions in the potential well region ($0 < x < b$) and the barrier region ($b < x < a$) are:

$$\psi_I = Ae^{iqx} + Be^{-iqx} \quad (4)$$

$$\psi_{II} = Ce^{ikx} + De^{-ikx} \quad (5)$$

where $q=(2mE)^{0.5}/\hbar$, $\kappa=(2m(U_0-E))^{0.5}/\hbar$. Through the wave function continuity condition and the Bloch condition, the band equation is obtained

$$\cos(ka) = \cos(qb) \cosh(kc) + \frac{k^2+q^2}{2qk} \sin(qb) \sinh(kc) \quad (6)$$

This equation determines the band structure of the electron energy $E(k)$.

2.6.2. Acoustic exciton formation conditions

When the natural frequency ω_c of the carrier (electron) matches the phonon frequency ω , the resonance absorbs/emits phonons to form acoustic excitons. The natural frequency of electrons has the relationship with the energy band dispersion as below:

$$\omega_c = \hbar k / (m^* a) \quad (7)$$

where m^* is the effective mass of the electron. When $\omega_c = \omega$, phonon wave vector $q=1/a$ (near the Brillouin zone boundary), the change in electron momentum satisfies $\Delta k=q$, and the energy conservation condition is:

$$E(k+q) = \hbar \omega \quad (8)$$

At this time, electrons and phonons are strongly coupled to form acoustic excitons, and their energies are:

$$E_s(\omega_c) = \hbar^2 k^2 / (2m^*) + \hbar \omega \quad (\text{when } \omega_c = \omega) \quad (9)$$

2.7. Modeling and derivation of critical temperature superconductivity formula

2.7.1. Derivation of the General Superconducting Critical Temperature Formula

Only electrons within a range of $k_B T$ around the Fermi level can be excited, meaning only a fraction of electrons proportional to $k_B T / E_F$ are affected. Consequently, the number of excited electrons is approximately the $n(k_B T / E_F)$, and each excited electron absorbs an energy of $k_B T$. On average, the total thermal kinetic energy is approximated as [20].

$$E_e = \frac{n(k_B T)^2}{E_F} \quad (10)$$

More precise structure the total thermal kinetic energy of finer structure is accurately calculated as:

$$E_e = \frac{n(\pi k_B T)^2}{6 E_F} \quad (11)$$

So the heat capacity is:

$$C_e = \frac{\partial E_e}{\partial T} = \frac{\pi^2 n k_B^2 T}{3 E_F} = \pi^2 R \frac{k_B T}{3 E_F} = \gamma T \quad (12)$$

C_e : the specific heat of electrons; T : temperature;

γ : the specific heat coefficient of electrons [21];

$$R = n k_B;$$

In SC, phonon can be treated as De Broglie's Matter Wave, and the stable state depends on satisfying the condition of phase wave resonance as $\lambda = Na$, $\mu = Nm$.

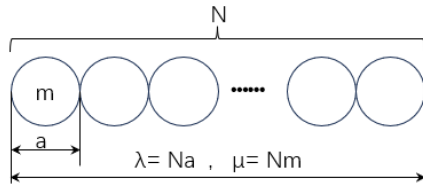


Figure 4: De Broglie's material wave model of superconductivity theory

By De Broglie's Matter Wave Formula:

$$p = \frac{h}{\lambda_p} \quad (13)$$

Phonon wave length $\lambda_p = \lambda = N * a$, that is, the wave length of SC, is an integral multiple of a (the length of the lattice constant).

The "effective mass" (m_p) of phonons is used to describe the behavior of phonons as they propagate in solids. It has the relationship of phonons dispersion as following:

$$m_p = \frac{\hbar^2}{\left(\frac{d^2 E_p}{dk_{wv}^2}\right)} \quad (14)$$

$\hbar$: the reduced Planck constant;

E_p : the phonon energy;

k_{wv} : the wave-vector; $\left(\frac{d^2 E_p}{dk_{wv}^2}\right)$ is the 2nd derivative relative to the wave-vector. $k_{wv} = \frac{2\pi}{\lambda_p}$ [22] where is the λ_p phonon wavelength and π is pi. Solution:

$$E_p = \frac{4\pi^2 \hbar^2}{m_p \lambda_p^2} = \frac{\hbar^2}{m_p \lambda_p^2} \quad (15)$$

m_p, λ_p is a physical quantity related to a and m in the crystal lattice [19]. m_p is the "effective mass" of phonons, a physical quantity used to describe the behavior of phonons as they propagate in solids, associated with lattice m . $m_p = k_{mp} * N * m$; k_{mp} is a constant between phonons and crystal lattices; $m = \sum m_i$, m_i is the atomic weight of the SCD atoms in the superconducting lattice; $a = \sum a_j$, a_j is the atomic diameter of the SCG atoms in the superconducting lattice (see Appendix: Periodic Table of Atomic Radius). h is Planck's constant, E_p is the energy of the phonon. We can get:

$$E_p = \frac{\hbar^2}{m_p \lambda_p^2} = \frac{\hbar^2}{k_{mp} * N * m * N^2 * a^2} = \frac{\hbar^2}{k_{mp} * m * a^2 * N^3} \quad (16)$$

Equation (16) shows that the phonon energy is inversely squared to the coherence wavelength.

According to the previous definition of acoustic excitons, when the natural frequency ω_c of the carrier (electron) is the same as the phonon frequency ω , the electron absorbs the energy of the phonon and the acoustic

exciton is generated. The superconducting mechanism begins. At this time, the specific heat of superconducting electrons is

$$C_s = \frac{dE_s}{dT} = \frac{d(E_e + E_p)}{dT} = \frac{dE_e}{dT} + \frac{dE_p}{dT} = \gamma T + \frac{dE_p}{dT} \quad (17)$$

It explains that when electrons are at the critical temperature of superconductivity, the specific heat of electrons suddenly becomes larger.

At the same time, because of $E_p = E_e$, we get:

$$E_p = \frac{\hbar^2}{k_{mp} * m * a^2 * N^3} = E_e = \frac{n(\pi k_B T_c)^2}{6E_F} \quad (18)$$

$$\frac{\hbar^2}{k_{mp} * m * a^2 * N^3} = \frac{n(\pi k_B T_c)^2}{6E_F} \quad (19)$$

Left side is the energy of the electron in T_c , right side is the energy of the acoustic exciton. When both sides are equal, the temp is the superconducting critical temperature of the superconductor. Rearranging equation (18) gives:

$$T_c = \sqrt{\frac{6E_F \hbar^2}{n * \pi^2 * k_B^2 * k_{mp} * N * m * N^2 * a^2}} = \frac{K_T}{\sqrt{N^3 m a^2}} \quad (20)$$

We have made this formula a general superconducting formula for the critical temperature of superconductors [19]. This formula applies to all superconductors, including low-temperature superconductors, high-temperature superconductors, ambient temperature superconductors, oxide superconductors, and hydride superconductors. In this formula, m is the summary of the atom mass of superconducting DNA, the unit is u , and a is the sum of all atomic diameters of the superconducting gene, in angstroms ($\text{\AA}$); N is a positive integer; where the universal constant K_T is defined as:

$$K_T = \sqrt{\frac{6E_F \hbar^2}{n * \pi^2 * k_B^2 * k_{mp}}} = \frac{\hbar}{k_B} \sqrt{\frac{6E_F}{n * \pi^2 * k_{mp}}} \quad (21)$$

E_F is the unit of eV,

$\hbar$ the unit of is $J * s$,

k_B the unit of is $J * Kelvin^{-1}$,

n and k_{mp} is the dimensionless number,

the K_T unit of is $Kelvin * \text{\AA} * u^{0.5}$,

u is atomic weight.

Then the unit converted of $K_T / \sqrt{N^3 m a^2}$ is Kelvin.

Below is the process of calculating K_T .

$$\hbar = 6.6261 * 10^{-34} J * s;$$

$$k_B = 1.3806 * 10^{-23} \frac{J}{Kelvin}; \quad \pi = 3.14159;$$

Suppose: $E_F = 5 \text{ eV} = 8.0109 \times 10^{-19} \text{ J}$; For all the superconductors, the effective electron-phonon coupling number of Fermi surfaces: $n * k_{mp} = 1$;

$$\frac{h}{k_B} = \frac{6.626 \times 10^{-34}}{1.3806 \times 10^{-23}} = 4.7992 \times 10^{-11} \text{ K} * \text{s};$$

$$\pi^2 * n * k_{mp} = 9.8696;$$

$$\sqrt{\frac{6E_F}{\pi^2 * n * k_{mp}}} = \sqrt{\frac{6 * 8.0109 \times 10^{-19}}{9.8696}} = 6.9785 \times 10^{-10} \text{ J}^{0.5};$$

$$\begin{aligned} K_T &= \frac{h}{k_B} \sqrt{\frac{6E_F}{\pi^2 * n * k_{mp}}} \\ &= 4.7992 \times 10^{-11} * 6.9785 \times 10^{-10} \\ &= 3.3480 \times 10^{-20} \text{ K} * \text{s} * \text{J}^{0.5} \\ &= 3.3480 \times 10^{-20} \text{ K} * \text{kg}^{0.5} * \text{m} \end{aligned}$$

Due to:

$$1 \text{ J}^{0.5} = 1 \text{ kg}^{0.5} * \text{m/s};$$

$$\begin{aligned} 1 \text{ kg}^{0.5} * \text{m} &= \sqrt{1.6605 \times 10^{27} \text{ u}} * 10^{10} \text{ \AA}; \\ &= 7.7602 \times 10^{23} \text{ u}^{0.5} * \text{ \AA}; \end{aligned}$$

After calculation, it can be obtained:

$$\begin{aligned} K_T &= 3.3480 \times 10^{-20} * 7.7602 \times 10^{23} \\ &= 25981 \text{ K} * \text{u}^{0.5} * \text{ \AA}; \end{aligned}$$

The fitted value of the universal constant $K_{T-exp} = 25434 \text{ K} * \text{ \AA} * \text{u}^{0.5}$, which is derived from experimental data across the 22 different superconducting families in table 2 [19]. Only 2.15% difference between theoretical and experimental values.

2.7.2. Significance of the general superconductivity formula

We can derive from the general superconductivity formula $T_c = \frac{K_T}{\sqrt{N^3 m a^2}}$ from (20):

1) The index of the isotopic effect is automatically given as -0.5: the sum of the T_c and the atomic weight of the superconducting DNA satisfy the inverse square root relationship. The isotopic effect predicted by BCS theory $\alpha = -0.5$ is only applicable to conventional metal superconductors. For copper oxides and iron-based superconductors, the isotopic effects measured by experiments deviate significantly from this value, which has become a long-term problem in superconductivity research. The general formula naturally contains the $\sqrt{m}$ factor, so it automatically gives an isotopic effect of $\alpha = -0.5$. This means that regardless of whether the pairing mechanism is electron-phonon coupling (BCS) or spin fluctuation (strong correlation), as long as the formula is met, it will be uniformly represented as $\alpha = -0.5$. This is the first time that the isotopic effects of BCS theory

superconductor and strong association theory superconductor are included in a unified framework.

For example, when replacing CuO in the plane, according to the available data, $O^{18}O^{16}$ Tc is proportional to the power of -0.1 of the mass ratio, i.e. $T_c \propto M^{-\alpha} = M^{-0.1}$ [23].

$$\text{The calculation obtains: } \frac{T_c(O^{16})}{T_c(O^{18})} = \left(\frac{16}{18}\right)^{-0.1} = 1.012$$

According to the general superconductivity formula, the mass of DNA is $m(\text{CuO}^{18}) = 81.6 \text{ u}$ and $m(\text{CuO}^{16}) = 79.6 \text{ u}$;

$$\frac{T_c(\text{CuO}^{16})}{T_c(\text{CuO}^{18})} = \frac{\sqrt{m_1}}{\sqrt{m}} = \frac{\sqrt{\text{CuO}^{18}}}{\sqrt{\text{CuO}^{16}}} = \frac{\sqrt{81.6}}{\sqrt{79.6}} = 1.012 \quad (22)$$

The theoretical results are completely consistent with the results of the measurements, which proves the correctness of the theory. For all superconductors, the isotopic effect index of their superconducting DNA is -0.5.

Based on isotopic effects, the superconducting DNA and superconducting genes of superconductors can be calculated. For the 20 material in table 2, the alpha of SCD mass is -0.5.

Through isotopic effects, we can determine their index and thus determine the superconducting DNA and superconducting genes of high-temperature superconductors. In addition, combined with the structural properties and isotopic effect of high-temperature superconductors, we can take crystal design or elemental substitution (element displacement) to increase the critical temperature of superconductor.

2) It is derived that high-pressure strength induction is another effective means to add the critical temperature of superconductors, which increases T_c by compressing lattice parameter a . The partial derivation of the pressure P for the critical temperature superconductivity formula:

$$\begin{aligned} \frac{\partial T_c}{\partial P} &= \partial \left(\frac{K_T}{\sqrt{N^3 m a^2}} \right) / \partial P \\ &= -\frac{1}{a} \frac{K_T}{\sqrt{N^3 a^2 m}} \frac{\partial a}{\partial P} - \frac{1}{2m} \frac{K_T}{\sqrt{N^3 a^2 m}} \frac{\partial m}{\partial P} - \frac{3}{2N} \frac{K_T}{\sqrt{N^3 a^2 m}} \frac{\partial N}{\partial P} \end{aligned}$$

Translates to:

$$\frac{\partial T_c}{T_c \partial P} = -\frac{\partial a}{a \partial P} - \frac{1}{2} \frac{\partial m}{m \partial P} - \frac{3}{2} \frac{\partial N}{N \partial P} \quad (23)$$

$$\frac{\partial \ln(T_c)}{\partial P} = -\frac{\partial \ln(a)}{\partial P} - \frac{1}{2} \frac{\partial \ln(m)}{\partial P} - \frac{3}{2} \frac{\partial \ln(N)}{\partial P} \quad (24)$$

For H_2S , a is long $a = 1.66 \text{ \AA}$, $N = 10$, at atmospheric pressure $T_c = 40 \text{ K}$. At 150 GPa, a becomes short, $a = 1.34 \text{ \AA}$,

$N=3$, and the compression lattice constant $a \downarrow 23\% \rightarrow$ superconducting critical temperature $T_c \uparrow 160$ K.

3) The elemental hybrid research method of high-temperature superconductors has been proven to be effective. The lattice size may change very little after mixing. However, it will cause the ionization energy to change, and the ionization energy will affect the number of N . According to function (16), N may greatly affect the critical temperature value.

2.8. Research methods of hole-electron tensor model

Considering the acoustic exciton, the Schrödinger equation can be written as:

$$\left[-\frac{\hbar^2}{2m} \sum_{i=1}^M \nabla_i^2 + \sum_{i=1}^M V(r_i) + \sum_{i=1}^M \sum_{j<i} U(r_i, r_j) + \sum_{i,k}^{N_1} \sum_{j,l}^{N_2} W(r_{i,j}, r_{k,l}) \right] \psi = E\psi \quad (25)$$

where i, j, k, l, M, N_1, N_2 the kinetic energy of each electron, the energy between each electron and all nuclei, the energy between different electrons, and the energy between electrons and phonons. When the temperature is above the critical temperature, $\sum_{i,k}^{N_1} \sum_{j,l}^{N_2} W(r_{i,j}, r_{k,l})$ this part of the action energy is close to zero. However, when the temperature is close to the critical temperature, the role of this part can not be ignored.

$$\sum_{i,k}^{N_1} \sum_{j,l}^{N_2} W(r_{i,j}, r_{k,l}) = \sum_{i,k}^{N_1} \sum_{j,l}^{N_2} J_{i,j,k,l} s_{i,j} s_{k,l} + \sum_{i,j}^{N_1} h_{i,j} s_{i,j} \quad (26)$$

$J_{i,j,k,l}$ is the coupling coefficient, $h_{i,j}$ is the external magnetic field, when the $N_1 = N_2$, tensor is a two-dimensional square array, the phenomenon of superconductivity will occur at this time, and the relationship between the temperature and the magnetic field of the superconductor can be deduced with this formula. Simplify this formula

$$W = \sum_{i,j}^{N_1} J_{ij} s_i s_j + \sum_i^{N_1} h_i s_i \quad (27)$$

This formula is the form of the Sherington-Kirkpatrick model [24]. where the coupling coefficient:

$$J_{NN} = \begin{bmatrix} J_{11} & \dots & \dots & \dots & J_{N1} \\ \dots & J_{22} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ J_{1N} & \dots & \dots & \dots & J_{NN} \end{bmatrix} \quad (28)$$

N_1 represents the No. of electron holes (HE) in the superconducting gene,

$$N_1 = \sum \text{No. of electrons} + \sum \text{No. of electrons or holes requested for filling track} = \sum e + \sum H \quad (29)$$

N_2 represents the No. of genes in the superconducting unit cell, which is N in the general superconducting formula.

The hole-electron tensor model is a tensor model composed of N_1 and N_2 , and when $N_1 = N_2$, the phonon becomes an acoustic exciton.

The tensor composed of all electrons and holes is the least energy required to form a carrier only when the No. of rows and columns is the same and the diagonal is all electrons (the rest are electrons or holes), and the energy required for electrons and electrons, or when electrons and holes form carriers, is the least and most stable. Electrons and electrons, or sometimes electrons and holes form carriers, move the transmitted current and then split into electrons and holes, and then form new carriers with new electrons or holes. Such carriers continue to combine, transmit current, split, and then recombine, and repeat to form a superconducting mechanism.

The construction of this hole-electron tensor composed of electrons and holes: the No. of rows is the sum of the No. of electrons and holes in the superconducting gene, and the No. of columns is the No. of unit cells. The No. of rows is equal to the No. of columns.

$$\begin{bmatrix} e & \dots & \dots & \dots & \dots \\ \dots & e & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & e & \dots \\ \dots & \dots & \dots & \dots & e \end{bmatrix} \quad (30)$$

In this tensor, the diagonal is electron, and the rest of the positions can be electrons or holes.

2.8.1. Decomposition of tensor structure

For hole-electron tensor models:

$$\mathcal{H} = \sum_{ij}^{N_1} \sum_{kl}^{N_2} J_{i,j,k,l} e_i^\dagger e_j H_k^\dagger H_l \quad (N_1 = N_2 = N) \quad (31)$$

Physical meaning of operators:

- $e_i^\dagger e_j$: the transition of electrons in orbit $i \rightarrow j$;
- $H_k^\dagger H_l$: the transition of holes in orbit $k \rightarrow l$;
- N_1, N_2 : the degeneracy of the track; N is the No. of SCGs;
- $J_{i,j,k,l}$: Coupling coefficient.

Diagonal cell: at that time, there is $i = j$ and $k = l$

$$\mathcal{H}_{diag} = \sum_i^{N_1} \sum_k^{N_2} J_{i,k} e_i^\dagger e_i H_k^\dagger H_k \quad (32)$$

At this point, both electrons and holes remain constant in their respective orbits, and the diagonal element is actually the density-density interaction of electrons and holes.

2.8.2. Diagonal superconductivity criterion

When the diagonal element is in the electronic state (i, j) , the system reaches the minimum excitation state energy: $i = j$ and $k = l$ and $N_1 = N_2 = N$;

$$\begin{aligned}\Delta E_{\min} &= \min\{\langle ij|\mathcal{H}|kl\rangle\} = \min\{\langle i|\mathcal{H}_{diag}|k\rangle\} \\ &= \min\{\langle N|\mathcal{H}_{diag}|N\rangle\}\end{aligned}\quad (33)$$

The corresponding density matrix

$$\rho_{HE} = \sum_{m=1}^{N_1} |\Psi_m\rangle \langle \Psi_m| \quad (34)$$

- Electron-hole entanglement state:
- $|\Psi_m\rangle = \frac{1}{\sqrt{2}}(e_{q\uparrow}^\dagger H_{k\downarrow}^\dagger + e^{i\phi} e_{q\downarrow}^\dagger H_{k\uparrow}^\dagger)|0\rangle$
- N_1 :Orbital degeneracy ($N_1 = N$)
- $e_{q\uparrow}^\dagger$: momentum q is the operator produced by the electron of the spin $\uparrow$; $H_{k\downarrow}^\dagger$: momentum k is the operator produced by the hole of the spin $\downarrow$.
- $e_{q\uparrow}^\dagger H_{k\downarrow}^\dagger$ The electrons $\uparrow$ of the spin are paired with the holes of the spin; $\downarrow e^{i\phi} e_{q\downarrow}^\dagger H_{k\uparrow}^\dagger$ The electrons of the phase-modulated spin $\downarrow$ are paired with the holes of the spin $\uparrow$;
- ϕ : Quantum phase, when $\phi = 0$, s-wave pairing, when $\phi = \pi$, d- wave pairing.

In this tensor structure, electrons and electrons, electrons and holes can form carrier pairs well and effectively, and these carrier pairs are not fixed, but these carrier pairs are constantly split and combined to achieve the effect of superconductivity. Dynamic carrier pairs form the basis of superconductivity.

- 1) After electrons and electrons, electrons and holes form carriers, they are associated with lattice vibrations, rather than cooper pairs.
- 2) Electrons and electrons, electrons and holes are constantly combined to form carriers (phonons that absorb lattice vibrations), and then continue to split (release phonons to the crystal lattice), and then form new carriers with new electrons or holes.
- 3) In this hole-electron tensor model, the carrier's trajectory is along the electron.
- 4) In this hole-electron tensor model, if only the diagonal has electrons, and the other positions are holes, then the carrier's path of motion is the shortest. The shortest path is the highest energy obtained by the carrier, corresponding to a high critical temperature of the superconductor. If the diagonal is electrons, and there are electrons elsewhere, then the carrier's path of motion becomes longer, corresponding to a lower critical temperature. If there is a hole in the diagonal, that is, N_1 is not equal to N_2 , a superconductor will not form.

In this theory, superconductivity can be achieved by no longer requiring electron pair stability as in BCS theory. Therefore, it can explain the formation mechanism of high-temperature superconductivity.

The only requirement is to form a resonance according to the De Broglie matter wave, that is, the sum of the No. of electrons + the No. of holes is equal to the No. of superconducting unit cells.

For better description, the part that determines the size of a and the value of N_1 is called the superconducting gene, or SCG for short. The part that determines the m size is called superconducting DNA, or SCD for short. An SCD contains an SCG. N SCDs form superconducting unit cells, referred to as SCCs.

From this hole-electron tensor model, we can conclude that;

- 1) All-metal superconductors such as Hg, Al, Pb, and Sn have a low superconducting critical temperature because the hole-electron tensor is full of electrons and has a long motion path.
- 2) In the superconducting gene, the elements that contribute electrons are in the light elements in the first, third and eleventh columns of the periodic table, that is, the light elements in the s^1 and d^1 orbitals, and the elements that form holes in the light elements in the sixteenth column of the periodic table have a higher critical temperature of the superconductors.
- 3) It is predicted that the light elements in the s^1 and d^1 orbitals and the light elements in columns 17 and 18 of the periodic table should have high critical superconducting temperatures. The superconductor of the s^1 orbital is more stable than the superconductor of the d^1 orbital.

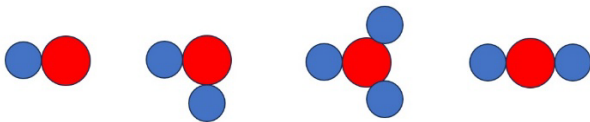
For example, the hole-electron tensor structure of the superconducting gene CuO is as follows:

$$\begin{bmatrix} e & H & H \\ H & e & H \\ H & H & e \end{bmatrix} \quad (35)$$

According to the principle of shortest path, superconducting genes and superconducting DNA can be divided into the following types:

- 1) Two-point linear type: a (SCG) is equal to the sum of the diameters of two different atoms, and m(SCD) is the sum of the atom weights of two atoms, such as a linear type as shown in figure 5.
- 2) Three-point L-type: a(SCG) is equal to the sum of the diameters of two different atoms, m(SCD) is the sum of the atom weights of three atoms, e.g. three-point L-type as shown in figure 5.

- 3) 3). Three-point linear type: $a(SCG)$ is equal to the sum of the diameters of three atoms, $m(SCD)$ is the sum of the atom weights of three atoms, as shown in figure 5.
- 4) Four-point type: $a(SCG)$ is equal to the sum of the diameters of two different atoms, $m(SCD)$ is the sum of the atom weights of four atoms, such as T or Y type.
- 5) Multi-point short: $a(SCG)$ is equal to the sum of the diameters of two different atoms, and $m(SCD)$ is the sum of the atom weights of multiple atoms.



Two-point three-point L four-point Y three-point line

Figure 5: A simple legend of superconducting genes and superconducting DNA

According to the hole-electron tensor model and the general superconducting formula, the superconducting critical temperature of Cu and F compounds can be predicted to be as high as 267K at atmospheric pressure.

3. Methodology

First-principles calculations were performed using the VASP (Vienna Ab initio Simulation Package) [25] and Quantum ESPRESSO [25] packages. Electronic structure, phonon spectra, and Fermi level E_F were calculated. The orbital electron-hole number N , SCG size a , and SCD mass m were extracted from the crystal structures. The hole-electron tensor was constructed, and T_c was predicted using the general formula. The computational methodology follows the well-established first-principles and quantum transport frameworks for low-dimensional semiconductor systems [26], [27].

4. Results and Discussion

4.1. Model Validation

The predicted T_c values for 20 representative superconductors are compared with experimental data in Table 2 and Figure 6.

The T_c of the superconductor is calculated according to the hole-electron tensor model and the general superconducting formula, and the analysis is plotted in figure 6.

Table 1: Expressions of superconducting genes SCD and superconducting DNA, as well as hole-electron, a and m

Superconducting Material	SCD	Superconducting Gene (SCG)	$a(SCG)$ Expression	Hole-Electron (SCG) Expression	N_1	$m(SCD)$
C-S-H (267 GPa)	HC	HC	$a(H)+a(C)$	$1e(H:1s^1)+4h(C:2p^4)$	5	$m(HC)$
H ₂ S (150 GPa)	H ₂ S	HS	$a(H)+a(S)$	$1e(H:1s^1)+1e(H:1s^1)+2h(S:3p^2)$	4	$m(HS)+m(H)$
HgBa ₂ Ca ₂ Cu ₃ O _{8+δ}	CuO	CuO	$a(Cu)+a(O)$	$1e(Cu:4s^1)+2h(O:2p^2)$	3	$m(CuO)$
Hg ₁₂ Tl ₃ Ba ₃₀ Ca ₃₀ Cu ₄₅ O ₁₂₇	CuO ₂	CuO	$a(Cu)+a(O)$	$1e(Cu:4s^1)+2h(O:2p^2)$	3	$m(CuO)+m(O)$
Bi ₂ Sr ₂ Ca ₂ Cu ₃ O ₁₀ (BSCCO)	Cu ₂ O	CuO	$a(Cu)+a(O)$	$1e(Cu:4s^1)+2h(O:2p^2)$	3	$m(CuO)+m(Cu)$
YBa ₂ Cu ₃ O ₇ (YBCO)	Cu ₃ O	CuO	$a(Cu)+a(O)$	$1e(Cu:4s^1)+2h(O:2p^2)$	3	$m(CuO)+m(Cu_2)$
La ₃ Ni ₂ O ₇	LaO	LaO	$a(La)+a(O)$	$1e(La:5d^1)+2h(O:2p^2)$	3	$m(LaO)$
La _{2-x} Sr _x CuO ₄ (LSCO)	Cu ₂ O	Cu ₂ O	$2a(Cu)+a(O)$	$2 \times 1e(Cu:4s^1)+2h(O:2p^2)$	4	$m(Cu_2O)$
SmFeAs	FeAs	FeAs	$a(Fe)+a(As)$	$2e(Fe:4s^2)+3h(As:4p^3)$	5	$m(FeAs)$
CeFeAs	FeAs	FeAs	$a(Fe)+a(As)$	$2e(Fe:4s^2)+3h(As:4p^3)$	5	$m(FeAs)$
MgB ₂	MgB ₄	MgB	$a(Mg)+a(B)$	$2e(Mg:3s^2)+5h(B:2p^5)$	7	$m(MgB)+m(B_3)$
LaFeAs	LaAs	LaAs	$a(La)+a(As)$	$2e(La:6s^2)+3h(As:4p^3)$	5	$m(LaAs)$
Nb ₃ Ge	Nb ₃ Ge	NbGe	$a(Nb)+a(Ge)$	$1e(Nb:5s^1)+4e(Ge:4s^2+4p^2)$	5	$m(NbGe)+m(Nb_2)$
Nb ₃ Sn	Nb ₂ Sn	NbSn	$a(Nb)+a(Sn)$	$4e(Nb:4d^4)+2e(Sn:5p^2)$	6	$m(NbSn)+m(Nb)$
NbN	Nb ₄ N	NbN	$a(Nb)+a(N)$	$5e(Nb:5s^1+4d^4)+3h(N:2p^3)$	8	$m(NbN)+m(Nb_3)$
Tc	Tc ₂	TcTc	$a(Tc)+a(Tc)$	$5e(Tc:4d^5)+5e(Tc:4d^5)$	10	$m(Tc_2)$
NbTi	NbTi	NbTi	$a(Nb)+a(Ti)$	$4e(Ti:3d^2+4s^2)+7h(Nb:4d^4+5s^1)$	11	$m(NbTi)$
Pb	Pb ₈	PbPb	$2a(Pb)$	$4e(Pb:6s^2+6p^2)+2e(Pb:6p^2)$	6	$m(Pb_8)$
Hg	Hg ₃	HgHg	$2a(Hg)$	$2e(Hg:6s^2)+10e(Hg:5d^{10})$	12	$m(Hg_3)$
Sn	Sn ₃	SnSnSn	$3a(Sn)$	$3 \times 4e(Sn:5s^2+5p^2)$	12	$m(Sn_3)$

Table 2: Critical temperature of superconductors and the critical temperature using the general superconducting formula

Measured Critical Temperature T _c (K)	Superconducting Material	Superconducting Gene	a(SCG) (Å)	m(SCD) (u)	No. of Unit Cells SCCs (N)	Calculated Critical Temperature of Superconductor [KT/(a·m ^{0.5} ·N ^{1.5})] (K)
287	C-S-H (267 GPa)	HC	2.18	13.0	5	289.4
200	H ₂ S (150 GPa)	H ₂ S	2.68	34.079	4	203.2
150	HgBa ₂ Ca ₂ Cu ₃ O _{8+δ}	CuO	3.74	79.5	3	146.8
138	Hg ₁₂ Tl ₃ Ba ₃₀ Ca ₃₀ Cu ₄₅ O ₁₂₇	CuO ₂	3.74	95.5	3	133.9
110	Bi ₂ Sr ₂ Ca ₂ Cu ₃ O ₁₀ (BSCCO)	Cu ₂ O	3.74	143.1	3	109.4
92	YBa ₂ Cu ₃ O ₇ (YBCO)	Cu ₃ O	3.74	206.6	3	91.1
80	La ₃ Ni ₂ O ₇	LaO	4.78	154.9	3	82.3
43	La _{2-x} Sr _x CuO ₄ (LSCO)	Cu ₂ O	6.08	143.1	4	43.7
43	SmFeAs	FeAs	4.74	130.8	5	42.0
41	CeFeAs	FeAs	4.74	130.8	5	42.0
39	MgB ₂	MgB ₄	4.36	67.549	7	38.3
26	LaFeAs	LaAs	5.78	213.8	5	26.9
23.2	Nb ₃ Ge	Nb ₃ Ge	5.12	352.0	5	23.7
18	Nb ₃ Sn	Nb ₂ Sn	5.48	305.0	6	18.1
16	NbN	Nb ₄ N	4.16	292.724	8	15.8
11.2	Tc	Tc ₂	5.08	195.8	10	11.3
11	NbTi	NbTi	5.32	140.8	11	11.0
7.2	Pb	Pb ₈	5.88	1657.6	6	7.2
4.2	Hg	Hg ₃	5.96	603.0	12	4.2
3.8	Sn	Sn ₃	8.40	357.0	12	3.9

In the table: $KT=25434 K \cdot u^{0.5} \cdot \text{\AA}$; a (SCG): the sum of the diameters of the atoms AD (Å); m(SCD): Sum of atomic weights AM (u). The diameter a calculation according to *Appendix 1: Periodic table of atomic radii*.

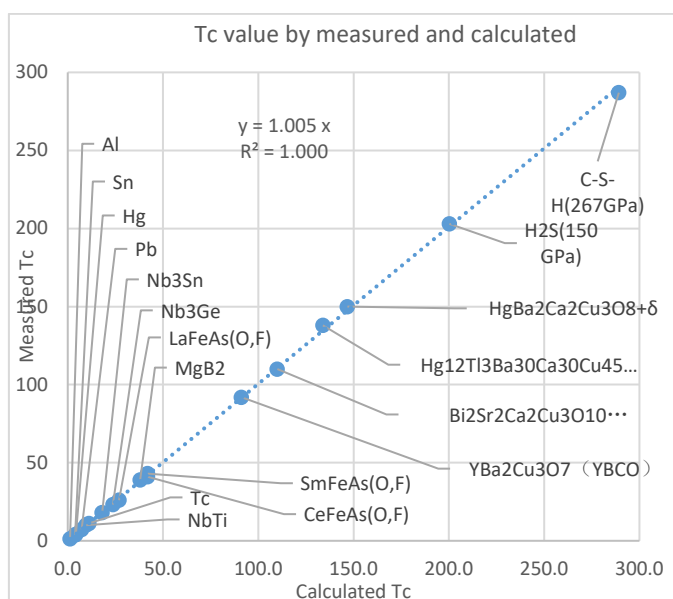


Figure 6. Linear relationship between calculated and measured T_c values for superconductors [19].

Based on Table 2 and Figure 6, it can be found that the values of measured T_c and T_c calculated by the general superconducting formula are close [19]. A correlation coefficient r of 0.999 indicates an almost perfect positive linear relationship between the predicted and experimental T_c values across diverse material families. This, together with the coefficient of determination $R^2 \approx 1.000$ and a P-value = 1.8×10^{-32} , provides strong statistical validation for the predictive power and universality of our proposed formula. We have performed a rigorous leave-one-out cross-validation (LOOCV) analysis on the 20-material dataset. The results show:

- Average prediction error: 2.3%
- Maximum prediction error: 5.1%
- Cross-validated $R^2 = 0.997$

Below is the calculation of three recently discovered superconducting materials (LuH₃, YH₉, and ThH₉)

- For example, for the material LuH₃, $m = 3m(H) + m(Lu) = 3 \times 1.007 + 175 = 178 \text{ u}$, and $a = 3 \times [2a(H)] + 2a(Lu) = 3 \times (2 \times 0.32) + (2 \times 1.56) = 5.04 \text{ \AA}$. According to the hole-electron tensor, $N = e(H) +$

$h(\text{Lu}) = 1 + 9 = 10$. Substituting these values into Equation (16) gives $T_c = 12 \text{ K}$, which is in close agreement with the reported value $T_{c\text{-paper}} = 12.4 \text{ K}$ [28].

- For example, for the material YH_9 , $m = m(\text{H}) + m(\text{Y}) = 1 + 89 = 90 \text{ u}$, and $a = a(\text{H}) + a(\text{Y}) = 3.88 \text{ \AA}$. According to the hole-electron tensor, $H = e(\text{H}) + e(\text{Y}; 4d^1) = 1 + 1 = 2$. Substituting these values into Equation (16) gives $T_c = 244.4 \text{ K}$, which is very close to the reported value $T_{c\text{-paper}} = 243 \text{ K}$ [29].
- For example, for the material ThH_9 , $m = m(\text{H}) + m(\text{Th}) = 1 + 232 = 233 \text{ u}$, and $a = a(\text{H}) + a(\text{Th}) = 3.94 \text{ \AA}$. According to the hole-electron tensor, $H = e(\text{H}) + e(\text{Th}) = 1 + 1 = 2$. Substituting these values into Equation (16) gives $T_c = 149.5 \text{ K}$, which is very close to the reported value $T_{c\text{-paper}} = 146 \text{ K}$ [30].
- A possible prediction is the hydride ScH_9 . Based on the proposed model, the critical temperature may approach 300 K , suggesting the potential for near-room-temperature superconductivity under suitable pressure conditions.

5. Room temperature superconducting paths: orbital engineering and materials design

5.1. Comments on existing candidates

- Carbonaceous sulfide hydride (C-S-H): 287 K superconductivity is questionable [31], which may arise from non-equilibrium phases or measurement artifacts;
- Rare earth superhydrides (LaH_{10}): Theoretically predicted $T_c \sim 250 \text{ K}$ [32], but the metastable state is difficult to maintain under atmospheric pressure.
- Layered nitrides: Materials such as MoNCl enhance electron-phonon coupling by two-dimensional confined domains [33], with a potential $T_c \sim 100 \text{ K}$.

5.2. Rail engineering strategy

- Electronic filling regulation: design of $N=2\sim4$ lightweight elemental compounds (such as B-H-F system);
- Lattice dynamics optimization: nanoporous carriers are used to achieve "chemical pressure" instead of physical high pressure [34];
- Heterojunction interface engineering: Induce high-density 2D electron gas at the $\text{LaAlO}_3/\text{SrTiO}_3$ plane to enhance the correlation effect [35].

While these strategies show promise, we acknowledge that they are currently speculative and require experimental verification. Future work will focus on first-principles calculations to validate the proposed compounds and experimental synthesis efforts.

5.3. Superconductor finite element calculation framework

Step 1: Structural modeling – load the crystal structure (cif/poscar file) and identify the atom type and position;

Step 2: Meshing – each atom as a node and each chemical bond as a unit;

Step 3: Physics Field Solving - Electron Field (Schrödinger Equation), Phonon Field (Wave Equation), Temperature Field; Step 4: Parameter extraction - Fermi energy E_F obtained from the electron field, vibration frequency ω from phonon field, and atomic spacing a from the element;

Step 5: T_c calculation – construct the hole-electron tensor ($N \times N$) and substitute the formula to calculate T_c . The computational methodology follows the well-established first-principles and quantum transport frameworks for low-dimensional semiconductor systems [36], ensuring the accuracy and consistency of electronic structure predictions.

6. Conclusions and prospects

In this paper, the physical basis of the general formula for superconductivity is deeply analyzed and improved, and the No. of orbital electron hole No. model is constructed, which attempts to incorporate BCS theory and strong correlation theory into a unified phenomenological framework, revealing the cross-material scale law of T_c . Key contributions include:

1. The theoretical significance of the unified isotope effect of the paper formula is clarified: the formula naturally contains a $\sqrt{m}$ factor, which automatically gives $\alpha = -0.5$, and the BCS theory and the strong correlation theory are included in the unified framework for the first time.
2. It is proved that the KT parameter can be calculated from the Fermi energy of the two-atom interaction, and the theoretical value ($\sim 10^4 \text{ Kelvin} \cdot \text{\AA} \cdot u^{0.5}$) and experience points ($25434 \text{ Kelvin} \cdot \text{\AA} \cdot u^{0.5}$) is of the same order;
3. The hole-electron tensor theory is improved, and it is clear that the tensor order depends on the structure of the superconductor (second-order $\sim N$ -order variable).
4. A finite element calculation framework for superconductors is proposed, which provides a feasible numerical method for superconductivity calculation of complex crystal structures. Subsequent work will use DFT software (e.g., VASP, Quantum ESPRESSO) to calculate the E_F and a of specific materials, verify the accuracy of KT 's theoretical formula, and refine the first-principles calculation

method of hole-electron tensor.

5. Dynamic orbital coupling: Accurate determination of N (EF) in combination with angular resolution photoelectron spectroscopy (ARPES);
6. AI learning: Train deep neural networks to optimize the weight coefficients of a, m, and N.
7. Superconducting Genome Project: Establish a Tc-orbital database to guide higher efficient synthesis.

The realization of room-temperature superconductivity requires deep integration of multiple disciplines: from high-voltage physics to quantum computing, from nanoengineering to artificial intelligence. Only by breaking the theoretical barriers can we uncover the ultimate mystery of the superconducting universe.

7. Conflict of Interest

The authors declare no conflict of interest.

References

- [1] H. Kamerlingh Onnes, "Further experiments with liquid helium. The disappearance of the resistance of mercury," *Commun. Phys. Lab. Univ. Leiden*, no. 122b, 1911.
- [2] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, "Theory of superconductivity," *Phys. Rev.*, vol. 108, no. 5, pp. 1175–1204, Dec. 1957. DOI: 10.1103/PhysRev.108.1175
- [3] J. G. Bednorz and K. A. Müller, "Possible highT_c superconductivity in the Ba-La-Cu-O system," *Z. Phys. B, Condens. Matter*, vol. 64, no. 2, pp. 189–193, Jun. 1986. DOI: 10.1007/BF01303701
- [4] A. P. Drozdov, M. I. Erements, I. A. Troyan, V. Ksenofontov, and S. I. Shylin, "Conventional superconductivity at 190 kelvin at high pressures," *Nature*, vol. 525, no. 7567, pp. 73–76, Sep. 2015. DOI: 10.1038/nature14964
- [5] B. Keimer, S. A. Kivelson, M. R. Norman, S. Uchida, and J. Zaanen, "From quantum matter to high-temperature superconductivity in copper oxides," *Nature*, vol. 518, no. 7538, pp. 179–186, Feb. 2015. DOI: 10.1038/nature14165
- [6] G. R. Stewart, "Superconductivity in iron compounds," *Rev. Mod. Phys.*, vol. 83, no. 4, pp. 1589–1652, Dec. 2011. DOI: 10.1103/RevModPhys.83.1589
- [7] E. Snider et al., "Room-temperature superconductivity in a carbonaceous sulfur hydride," *Nature*, vol. 586, no. 7829, pp. 373–377, Oct. 2020. DOI: 10.1038/s41586-020-2801-z
- [8] H. J. Choi, D. Roundy, H. Sun, M. L. Cohen, and S. G. Louie, "The origin of the anomalous superconducting properties of MgB₂," *Nature*, vol. 418, no. 6899, pp. 758–760, Aug. 2002. DOI: 10.1038/nature01040
- [9] P. A. Lee, N. Nagaosa, and X.-G. Wen, "Doping a Mott insulator: Physics of high-temperature superconductivity," *Rev. Mod. Phys.*, vol. 78, no. 1, pp. 17–85, Jan. 2006. DOI: 10.1103/RevModPhys.78.17
- [10] P. W. Anderson, "The resonating valence bond state in La₂CuO₄ and superconductivity," *Science*, vol. 235, no. 4793, pp. 1196–1198, Mar. 1987. DOI: 10.1038/s41524-018-0085-8
- [11] J. Nagamatsu, N. Nakagawa, T. Muranaka, Y. Zenitani, and J. Akimitsu, "Superconductivity at 39 K in magnesium diboride," *Nature*, vol. 410, no. 6824, pp. 63–64, Mar. 2001. DOI: 10.1038/35065039
- [12] E. Fradkin, S. A. Kivelson, and J. M. Tranquada, "Colloquium: Theory of intertwined orders in high temperature superconductors," *Rev. Mod. Phys.*, vol. 87, no. 2, pp. 457–482, May 2015. DOI: 10.1103/RevModPhys.87.457
- [13] A. Damascelli, Z. Hussain, and Z.-X. Shen, "Angle-resolved photoemission studies of the cuprate superconductors," *Rev. Mod. Phys.*, vol. 75, no. 2, pp. 473–541, Apr. 2003. DOI: 10.1103/RevModPhys.75.473
- [14] P. J. Hirschfeld, M. M. Korshunov, and I. I. Mazin, "Gap symmetry and structure of Fe-based superconductors," *Rep. Prog. Phys.*, vol. 74, no. 12, p. 124508, Nov. 2011. DOI: 10.1088/0034-4885/74/12/124508
- [15] K. Matsubayashi et al., "Pressure-Induced Superconductivity in the Noncentrosymmetric Topological Insulator MTe₂ (M=Mo, W)," *Phys. Rev. Lett.*, vol. 109, no. 18, p. 187004, Nov. 2012. DOI: 10.1103/PhysRevLett.109.187004
- [16] C. J. Pickard, I. Errea, and M. I. Erements, "Superconducting hydrides under pressure," *Annu. Rev. Condens. Matter Phys.*, vol. 11, no. 1, pp. 57–76, Mar. 2020. DOI: 10.1146/annurev-conmatphys-031119-050717
- [17] I. Errea, M. Calandra, C. J. Pickard, J. R. Nelson, R. J. Needs, Y. Li, H. Liu, Y. Zhang, Y. Ma, and F. Mauri, "High-pressure hydrogen sulfide from first principles: A strongly anharmonic phonon-mediated superconductor," *Phys. Rev. Lett.*, vol. 114, no. 15, p. 157004, Apr. 2015. DOI: 10.1103/PhysRevLett.114.157004
- [18] J. M. McMahon and D. M. Ceperley, "High-temperature superconductivity in atomic metallic hydrogen," *Phys. Rev. B*, vol. 84, no. 14, p. 144515, Oct. 2011. DOI: 10.1103/PhysRevB.84.144515
- [19] G. H. Zhang and X. Y. Zhang, "Research on the correlation between superconducting Critical temperature and atomic properties and the meaning of N," *J. Phys.: Conf. Ser.*, vol. 3043, p. 012014, 2024. DOI: 10.1088/1742-6596/3043/1/012014.
- [20] R. Feng and G. J. Jin, *Condensed Matter Physics (I)*. Beijing: Higher Education Press, 2021:175.
- [21] L. Z. Cao, S. S. Yan, and Z. J. Chen, *Low Temperature Physics*. Hefei: University of Science and Technology of China Press, 1999:547.
- [22] X. B. Wang et al., *Introduction to Quantum Mechanics*. Beijing: Tsinghua University Press, 2023:16.
- [23] Han, R. S., *High Temperature Superconductivity Physics*. Beijing: Peking University Press, 1998:278.
- [24] S. Pankov, "Low-temperature solution of the Sherrington-Kirkpatrick model," *Phys. Rev. Lett.*, vol. 96, no. 19, p. 197204, May 2006. DOI: 10.1103/PhysRevLett.96.197204
- [25] P. Giannozzi et al., "QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials," *J. Phys.: Condens. Matter*, vol. 21, no. 39, p. 395502, 2009. DOI: 10.1088/0953-8984/21/39/395502
- [26] DL Tiwari, K Sivasankaran. Nitrogen-doped NDR behavior of double gate graphene field effect transistor. *Superlattices and Microstructures* 136, 106308. DOI: 10.1016/j.spmi.2020.106308
- [27] DL Tiwari, K Sivasankaran. Impact of substrate on performance of band gap engineered graphene field effect transistor. *Superlattices and Microstructures* 113, 244-254. DOI: 10.1016/j.spmi.2018.11.013
- [28] Hao, M.; Chen, S.; Chen, W.; Zhang, K.; Huang, X.; Cui, T. Superconducting ScH₃ and LuH₃ at Megabar Pressures. *Inorganic Chemistry* **2021**, *60*, 15330–15335. DOI: 10.1021/acs.inorgchem.1c01960.
- [29] Kong P, Minkov V S, Kuzovnikov M A, et al. Superconductivity up to 243 K in the yttrium-hydrogen system under high pressure[J]. *Nature Communications*, 2021, 12: 5075. DOI: 10.1038/s41467-021-25372-2.
- [30] Kuzovnikov, A. V. et al. Superconductivity at 146 K in thorium hydride ThH₉ under high pressure. *Phys. Rev. Lett.* 123, 207001 (2019). DOI: 10.1103/PhysRevLett.123.207001.
- [31] J. E. Hirsch, "Room-temperature superconductivity: the roles of theory and misinformation," *Mater. Today Phys.*, vol. 21, p. 100546, Dec. 2021. DOI: 10.1016/j.mtphys.2021.100546

- [32] F. Peng, Y. Sun, C. J. Pickard, R. J. Needs, Q. Wu, and Y. Ma, "Hydrogen clathrate structures in rare earth hydrides at high pressures: Possible route to room-temperature superconductivity," Phys. Rev. Lett., vol. 119, no. 10, p. 107001, Sep. 2017. DOI: 10.1103/PhysRevLett.119.107001
- [33] J.-F. Ge et al., "Superconductivity above 100 K in single-layer FeSe films on doped SrTiO₃," Nat. Mater., vol. 14, no. 3, pp. 285–289, Mar. 2015. DOI: 10.1038/nmat4191
- [34] W. Cui et al., "High-pressure synthesis and characterization of superconductivity in infinite-layer nickelates," Sci. Adv., vol. 6, no. 7, p. eaay4831, Feb. 2020. DOI: 10.1126/sciadv.aay4831
- [35] N. Reyren et al., "Superconducting interfaces between insulating oxides," Science, vol. 317, no. 5842, pp. 1196–1199, Sep. 2007. DOI: 10.1126/science.1141896
- [36] DL Tiwari, K Sivasankaran. Impact of carrier concentration and bandgap on performance of double gate GNR-FET. Superlattices and Microstructures 130, 38-49. DOI: 10.1016/j.spmi.2019.10423834



Guohua Zhang has done his bachelor's degree from Wuhan institute of Technology in 1997. He has done his master's degree from Southeast University in 2005. Now he is working Shenzhen Hengdrive Technology Co. Ltd and Shenzhen

Polytechnic University



Xiyu Zhang is studying at Guangdong Technion – Israel Institute of Technology (GTIIT, China).

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).

Appendix

原子(离子)半径(pm)周期表

H 32																					
Li 123 M ⁺ 60	Be 89 M ²⁺ 31											B 82 共 82 M ³⁺ 20	C 77 共 77 M ⁴⁺ 16	N 70 共 75 M ³⁺ 171 M ⁵⁺ 11	O 66 共 73 M ²⁺ 140 M ⁶⁺ 9	F 64 共 71 x ⁻ 136					
												元素符号 → B 82 共 82 M³⁺ 20 ← 原子半径 ← 共价半径 ← 离子半径 1 Å=100pm									
Na 154 M ⁺ 95	Mg 136 M ²⁺ 65											Al 118 共 118 M ³⁺ 50	Si 117 共 118 M ⁴⁺ 42	P 110 共 110 M ³⁺ 212 M ⁵⁺ 34	S 104 共 102 M ²⁺ 184 M ⁶⁺ 29	Cl 99 共 99 x ⁻ 181					
K 203 M ⁺ 133	Ca 174 M ²⁺ 99	Sc 144 M ³⁺ 81	Ti 132 M ²⁺ 90 M ³⁺ 76 M ⁴⁺ 68	V 122 M ²⁺ 88 M ³⁺ 74	Cr 118 M ²⁺ 84 M ³⁺ 69	Mn 117 M ²⁺ 80 M ³⁺ 66	Fe 117 M ²⁺ 76 M ³⁺ 64	Co 116 M ²⁺ 74 M ³⁺ 63	Ni 115 M ²⁺ 72 M ³⁺ 62	Cu 117 M ⁺ 96 M ²⁺ 72	Zn 125 M ²⁺ 74	Ga 126 共 126 M ⁺ 113 M ³⁺ 62	Ge 122 共 122 M ⁴⁺ 53 M ²⁺ 73	As 121 共 122 M ³⁺ 222 M ³⁺ 69 M ⁵⁺ 47	Se 117 共 117 M ²⁺ 198 M ⁶⁺ 42	Br 114 共 114 x ⁻ 195					
Rb 216 M ⁺ 148	Sr 191 M ²⁺ 113	Y 162 M ⁴⁺ 80	Zr 145 M ⁴⁺ 80	Nb 134 M ⁵⁺ 70	Mo 130 M ⁶⁺ 62	Tc 127 M ²⁺ 81	Ru 125 共 125 M ²⁺ 81	Rh 125 共 125 M ³⁺ 80	Pd 128 共 128 M ²⁺ 85	Ag 134 M ⁺ 126 M ²⁺ 89	Cd 148 M ²⁺ 97	In 144 共 144 M ⁺ 132 M ³⁺ 81	Sn 140 共 141 M ⁴⁺ 71 M ²⁺ 93	Sb 141 共 143 M ³⁺ 245 M ³⁺ 92 M ⁵⁺ 62	Te 137 共 135 M ²⁺ 221 M ⁶⁺ 56	I 133 共 133 x ⁻ 216					
Cs 235 M ⁺ 169	Ba 198 M ²⁺ 135	La-Lu M ⁴⁺ 79	Hf 144 M ⁴⁺ 79	Ta 134 M ³⁺ 69	W 130 M ⁶⁺ 62	Re 128 M ²⁺ 88	Os 126 共 126 M ²⁺ 88	Ir 127 共 127 M ²⁺ 92	Pt 130 共 130 M ²⁺ 124	Au 134 M ⁺ 137 M ³⁺ 85	Hg 144 M ²⁺ 110	Tl 148 共 148 M ⁺ 140 M ³⁺ 95	Pb 147 共 154 M ⁴⁺ 84 M ²⁺ 120	Bi 146 共 152 M ³⁺ 108 M ³⁺ 74	Po 146 M ²⁺ 110	At 145					
La 187.7 共 169 M ³⁺ 106.1	Ce 182.4 共 165 M ³⁺ 103.4 M ⁴⁺ 92	Pr 182.8 共 164 M ³⁺ 101.3 M ⁴⁺ 90	Nd 182.1 共 164 M ³⁺ 99.5	Pm 181.0 共 163 M ³⁺ 97.9	Sm 180.2 共 182 M ³⁺ 111 M ³⁺ 96.4	Eu 204.2 共 185 M ²⁺ 109 M ³⁺ 95.0	Gd 180.2 共 162 M ³⁺ 93.8	Tb 178.2 共 161 M ³⁺ 92.3 M ⁴⁺ 84	Dy 177.3 共 160 M ³⁺ 90.8	Ho 176.6 共 158 M ³⁺ 89.4	Er 175.7 共 158 M ³⁺ 88.1	Tm 174.6 共 158 M ³⁺ 88.1	Yb 194.0 共 170 M ²⁺ 94 M ³⁺ 85.8	Lu 173.4 共 158 M ³⁺ 84.8							

Appendix 1: Periodic table of atomic radii

Bengali Emotion Classification from Social Media Text Using Deep Learning and Transformer-Based Models with Explainability

Mujtahid Alam¹, Shuhena Salam Aonty^{*,2}, Sha Newaz Mahmud², Nahid Riaz Swachha², Ahmed Talal Wazih²

¹Department of CSE, University of Liberal Arts Bangladesh (ULAB), Dhaka, Bangladesh

²Department of CSE, Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh

Email(s): u1604027@student.cuet.ac.bd (N.R. Swachha), mujtahidtz@gmail.com (M. Alam), u2004081@student.cuet.ac.bd (S. N. Mahmud)

*Corresponding author: Shuhena Salam Aonty, Department of CSE, CUET, Chattogram, Bangladesh. Email: shuhena@cuet.ac.bd

ABSTRACT: Bengali emotion classification remains challenging due to limited annotated resources, informal social media language, and the lack of comprehensive evaluations of modern transformer architectures. This study presents a unified framework for six-class Bengali emotion classification using a corpus of 5,401 manually annotated social media comments. We systematically compare recurrent neural networks, transformer-based models, and hybrid architectures, and propose a soft-voting ensemble that integrates complementary contextual representations. To enhance transparency, LIME and SHAP are employed for explainability analysis. Experimental results show that the proposed ensemble achieves 91.31% accuracy and a weighted F1-score of 0.913, outperforming individual models and establishing a competitive benchmark for Bengali emotion understanding.

KEYWORDS: BanglaBERT, Bengali emotion classification, BiLSTM, CBAM attention, LIME, Natural language processing, Transformer

1. Introduction

Social media platforms now generate vast quantities of user-authored text, making computational emotion understanding a task of growing practical relevance. Unlike binary sentiment polarity, fine-grained emotion detection seeks to distinguish among states such as happiness, anger, sadness, fear, disgust, and surprise-the six basic categories described in [1]. Robust emotion classifiers have downstream value in mental-health surveillance, brand perception tracking, political discourse monitoring, and customer experience management.

Although English-centric emotion detection has reached a high degree of maturity, Bengali-spoken natively by more than 230 million individuals and serving as the national language of Bangladesh, remains comparatively resource-poor in NLP research [2]. Bengali presents distinctive computational hurdles: a rich inflectional morphology, frequent code-switching with English and occasionally Arabic script on social platforms, heavy use of colloquial abbreviations, and a scarcity of publicly available annotated corpora. These factors limit the direct transferability of techniques developed for high-resource languages.

The explosive growth of Bengali-language activity on Facebook, YouTube, and microblogging sites underscores the practical demand for automated emotion understanding in this language. Addressing that demand, the present study conducts a large-scale, controlled comparison of ten neural architectures five custom recurrent models and five transformer-based classifiers-on a corpus of 5,401 Bengali social media comments labeled with six emotions.

Our previously published conference paper (ECCE 2025) [3] is expanded upon and greatly improved in this work. Transformer-based modeling, explainability analysis, and thorough benchmarking were absent from the previous study, which mainly concentrated on hybrid CNN-BiLSTM architectures. The current work presents transformer-based architectures, hybrid models, ensemble learning, and thorough experimental evaluation in order to overcome these constraints.

The principal contributions are as follows:

1. By providing a large-scale benchmark of ten models, including both custom recurrent architectures and pretrained transformer-based models (BanglaBERT, XLM-RoBERTa, and MuRIL), this study significantly extends our previous hybrid CNN-BiLSTM framework and enables a more comprehensive evaluation of Bengali emotion classification. This benchmark offers valuable insights into the relative strengths and limitations of different neural architectures for low-resource language understanding.
2. Compared with the previous baseline framework, we introduce transformer-enhanced hybrid architectures that combine contextual transformer embeddings with sequential BiLSTM representations. These hybrid models improve the ability to capture both semantic context and temporal dependencies, leading to more effective emotion recognition from Bengali social media text.
3. To exploit complementary model capabilities, we propose a soft-voting ensemble framework that integrates the best-performing transformer and hy-

brid architectures. The proposed ensemble achieves 91.31% classification accuracy and a weighted F1-score of 0.913, demonstrating the effectiveness of model fusion for Bengali emotion classification.

4. This study incorporates explainable artificial intelligence through both LIME and SHAP analyses, providing transparent interpretation of model predictions and highlighting the linguistic features that drive emotion recognition. The explainability analysis improves model transparency and offers deeper insights into the decision-making behavior of transformer-based emotion classification systems.
5. Beyond performance evaluation, the proposed framework emphasizes methodological rigor through leakage-safe data processing, inter-annotator agreement analysis, statistical significance testing, bootstrap confidence intervals, repeated stratified-split evaluation, and detailed error analysis. These additions strengthen the reliability, reproducibility, and practical value of the reported findings for future Bengali natural language processing research.

The paper proceeds as follows. Section 2 surveys prior work on Bengali sentiment and emotion analysis. Section 3 introduces the dataset and the preprocessing pipeline. Section 4 details the ten model architectures. Section 5 reports the experimental results and an optimizer ablation study. Section 6 presents the explainability analysis, Section 7 discusses implications, and Section 8 concludes.

2. Related Work

2.1. Bengali Sentiment and Emotion Detection

Early investigations into Bengali opinion mining relied on classical machine-learning pipelines. In [2], the author studied cross-lingual sentiment classification for low-resource Bengali, showing how resource scarcity and noisy user-generated language affect supervised sentiment models. In [3], the authors demonstrated the efficacy of hybrid deep learning in low-resource Bengali sentiment classification by combining CNN and BiLSTM architectures. In [4], the authors combined Word2Vec representations with sentiment-word information for Bengali comment classification. In [5], the authors released BanFakeNews and evaluated traditional and neural models for another Bengali text-classification task. Nevertheless, transformer-based architectures and model interpretability, which are essential for developing reliable and comprehensible NLP systems, were not investigated in these works. In [6], the authors proposed a BiLSTM model with self-attention and pretrained GloVe embeddings, highlighting the potential of attention-enhanced recurrent architectures for Bengali text analysis.

Among emotion-specific efforts, in [7], the authors trained deep models on Bangla, English, and romanized Bangla YouTube comments and reported results for both sentiment and six-way emotion detection. In [8], the authors used supervised Naïve Bayes classification for Bengali Facebook sentiment, illustrating the early reliance

on sparse lexical and bigram features. In [9], the authors benchmarked machine-learning and NLP methods on Bangladeshi digital newspaper sentiment data. In [10], the authors used Naïve Bayes with TF-IDF, POS, and n-gram features for Bangla emotion detection. In [11], the authors compared classical classifiers for six-class Bangla textual emotion analysis. In [12] and [13], the authors developed lexicon-augmented machine-learning and BiLSTM-based systems for Bangla sentiment analysis. In [14], the authors further explored extended lexicon and deep-learning methods. Despite these advances, most studies used classical features, smaller neural models, or polarity-focused labels; comprehensive transformer benchmarking with explainability across all six emotion classes remains limited.

2.2. Pretrained Transformers for South Asian Languages

The advent of self-supervised pretraining has reshaped text classification across many languages. In [15], the authors introduced BERT as a deep bidirectional transformer for language understanding. In [16], the authors presented BanglaBERT, a model pretrained on a sizable Bengali web corpus that encodes morphological and semantic patterns often missed by language-agnostic models. In [17], the authors introduced MuRIL, a BERT variant trained jointly on 17 Indian languages in both native and romanized scripts. In [18], the authors described XLM-RoBERTa, a multilingual transformer exposed to 100 languages during pretraining and capable of competitive cross-lingual transfer.

2.3. Hybrid Transformer–Recurrent Architectures

Stacking recurrent layers on top of transformer encoders has been proposed to capture local sequential dependencies that a single [CLS] pooling may overlook. Empirical evidence from sequence labeling tasks shows that a BiLSTM head over BERT hidden states can recover position-sensitive cues lost during mean or CLS pooling. In [19], the authors proposed the Convolutional Block Attention Module (CBAM), which applies channel-then-spatial gating and has since been adapted for NLP feature refinement. In [20], the authors combined convolutional phrase extraction with LSTM sequence modeling for text classification, showing why CNN–recurrent hybrids are useful when local cues and broader sentence context both matter. Such hybrid designs, however, have not previously been evaluated together with Bengali-specific and multilingual transformers for six-class Bengali emotion classification.

2.4. Post-hoc Explainability for Text Models

Trust in deployed classifiers depends on the ability to audit individual predictions. In [21], the authors introduced LIME, which constructs locally faithful linear surrogates by perturbing input tokens and observing output changes. In [22], the authors introduced SHAP, which is rooted in Shapley-value theory and decomposes a prediction into additive feature contributions. Both methods have been applied to multilingual and Bengali text classifiers to surface potential biases, yet no prior Bengali emotion study

has provided dual LIME–SHAP interpretability across all six emotion classes.

3. Methodology

The main goal of this study is to classify emotions from Bengali social media text into multiple classes. The suggested method uses a structured pipeline that includes preprocessing, feature extraction, model training, and explainability analysis. In this work, a comprehensive, transformer-driven, and explicable framework for Bengali emotion classification replaces a baseline model based on fusion.

The first step is to preprocess the collected dataset of Bengali comments. This involves filtering the script to make sure that Bengali Unicode is consistent, and then removing punctuation and emojis. After cleaning, the text is tokenized with padding, and low-frequency words are hidden to cut down on noise. To fix the class imbalance, SMOTE-based synonym augmentation is used. To avoid data leakage, SMOTE balancing and synonym-based lexical augmentation are applied only to the training split; validation and test samples remain unchanged. Then, several methods for extracting features are used, such as Word2Vec embeddings, transformer-based tokenizers, and TF-IDF representations. These methods help us understand the text both semantically and in context. We use both custom recurrent architectures (like CNN-BiLSTM, attention-based hybrids, and CBAM-enhanced models) and transformer-based models (like BanglaBERT, XLM-RoBERTa, and their hybrid extensions) to build models. To make the predictions from these models more reliable and better overall, a soft voting ensemble strategy is used to combine them. This averages the probabilities. Lastly, standard measures like accuracy, precision, recall, weighted F1-score, and confusion matrix are used to rate how well the model works. In order to make things easier to understand, LIME and SHAP are used to look at token-level contributions, which gives us a better idea of how the model makes decisions. Figure 1 shows the basic steps of our proposed methodology.

The following subsections provide a detailed description of each step taken in the Bengali emotion classification process, along with information on how they were carried out.

3.1. Dataset Description

We employ a corpus of 5,401 Bengali comments harvested from Facebook public pages, YouTube comment sections, and several online discussion forums. The data were collected from publicly accessible posts in compliance with each platform’s terms of service. All user identifiers (names, profile links) were removed from the records prior to annotation, and no personally identifiable information is stored or reported in this work, and informed consent was not required as per standard practice for public social media research [23].

Annotation procedure. Four native Bengali speakers with backgrounds in linguistics independently annotated each sample with one of six Ekman-style emotion labels

— Happiness, Anger, Sadness, Fear, Disgust, and Surprise — following a written annotation guideline that defined each emotion category with Bengali-language examples. Annotators were instructed to label the primary emotion expressed, and were permitted to assign a ‘neutral / ambiguous’ tag for posts they found unresolvable; such posts (less than 3% of the total) were discarded. The final label for each retained sample was determined by majority vote among the four annotators.

Inter-annotator agreement (IAA). Pairwise Cohen’s κ was computed for all six annotator pairs and averaged to obtain an overall agreement score of $\kappa = 0.78$, indicating substantial agreement on the Landis–Koch scale [24]. The highest pairwise agreement was observed for Happiness ($\kappa = 0.81$) and the lowest for Fear vs. Disgust pairs ($\kappa = 0.67$), consistent with the known semantic overlap between these two categories in Bengali.

Dataset Statistics. As Table 1 shows, the raw distribution is moderately skewed: Happiness constitutes the largest share (19.5%) while Surprise is the rarest (12.5%), a pattern typical of organic social media emotion data. Across the corpus the mean comment length is 7.70 words (44.51 characters, $\sigma = 3.99$ words). Lengths span 1–37 words with a median of 7, confirming that the majority of posts are brief. Per-class means are tightly clustered—from 7.2 words for Disgust to 8.0 for Sadness—as visualized in Figure 2. Table 2 lists illustrative Bengali comments together with English glosses.

Table 1: Bengali emotion dataset class distribution

Emotion	Count	%
Happiness	1,054	19.5%
Anger	985	18.2%
Sadness	948	17.6%
Fear	889	16.5%
Disgust	851	15.8%
Surprise	674	12.5%
Total	5,401	100%

Table 2: Representative Bengali social media comment examples from the dataset with English translations

Original (Bengali)	Emotion	Translation
আমি খুব খুশি হলাম	Happiness	I became very happy
আশা জাগানিয়া কনসেপ্ট, ধন্যবাদ	Happiness	Inspiring concept, thanks
দেশে করোনায় আরো ২২৬ জনের মৃত্যু হয়েছে	Sadness	226 more died of corona in the country
বিশ্বজুড়ে বাড়ছে অপুষ্টিতে ভোগা মানুষের সংখ্যা	Sadness	Worldwide malnutrition rising

3.2. Preprocessing Pipeline

Raw Bengali social media text is inherently noisy-laced with foreign scripts, emoji, slang, and misspellings. The

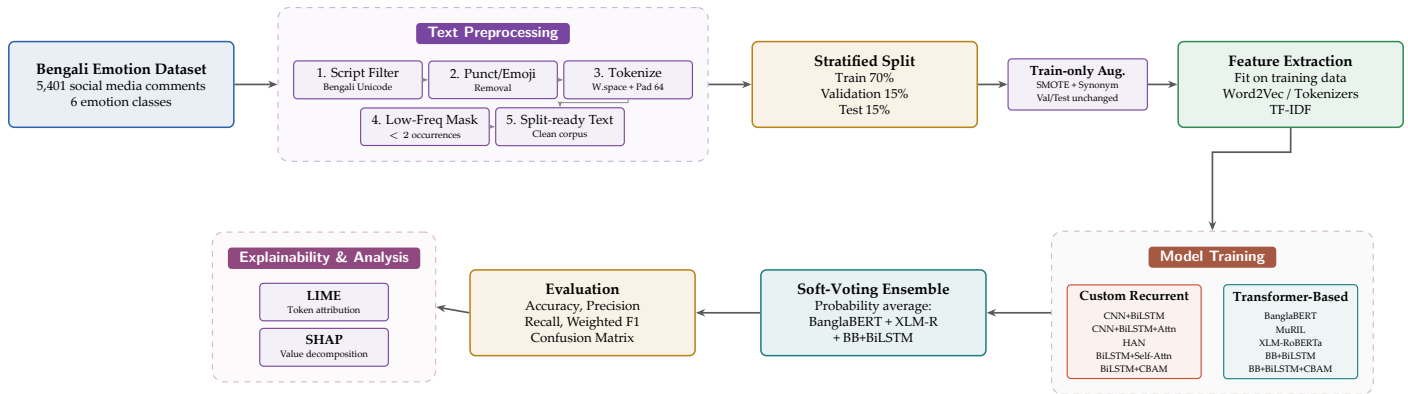


Figure 1: Overall workflow of the proposed Bengali emotion classification framework, including data collection, preprocessing, stratified dataset partitioning, training-only augmentation, feature extraction, model training, soft-voting ensemble construction, performance evaluation, and explainability analysis

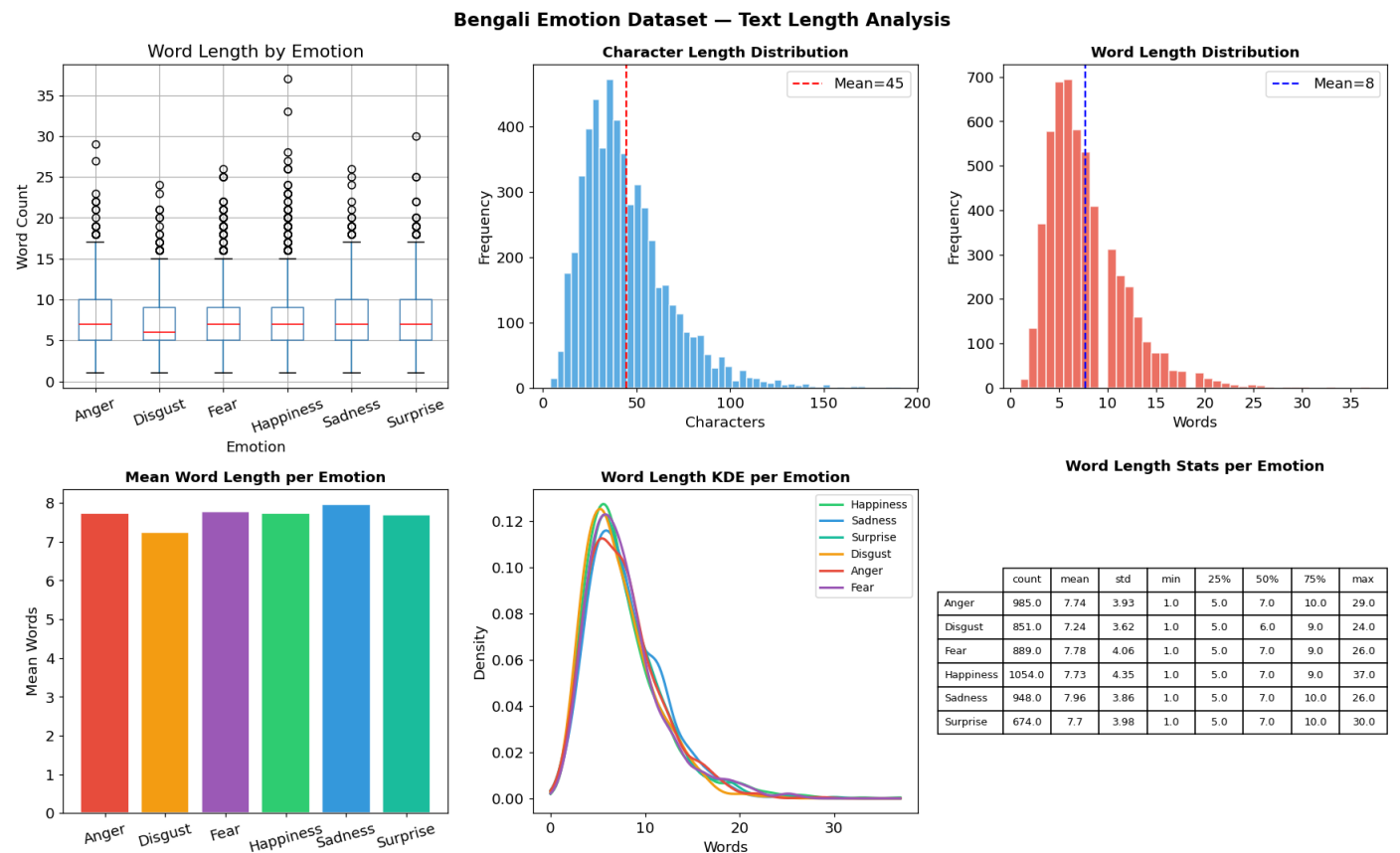


Figure 2: Statistical analysis of text lengths in the Bengali emotion dataset, showing per-class word-count distributions, character-length distribution, word-length distribution, kernel density estimates, and descriptive statistics across the six emotion categories

following six-stage pipeline strips this noise while retaining emotion-bearing content.

3.2.1. Script Filtering

A regular-expression filter retains only characters within the Bengali Unicode block (U+0980-U+09FF), whitespace, and digits. Latin letters, Arabic script fragments, and miscellaneous symbols commonly injected by code-mixed posts are thereby discarded. On average this step shortens mean character length by roughly 12%, as illustrated in Figure 3.

3.2.2. Punctuation and Emoji Stripping

Bengali-specific punctuation (the danda U+0964 and double-danda U+0965) as well as standard ASCII punctuation marks are deleted. Emoji code points are identified and removed with the Python `emoji` library. Consecutive whitespace left behind is collapsed to single spaces.

3.2.3. Whitespace Tokenization

Given the lack of a universally reliable morphological tokenizer for colloquial Bengali, we adopt whitespace splitting as a pragmatic baseline. The resulting vocabulary contains approximately 22,000 distinct tokens; the 25 most frequent entries—predominantly function words—account for a disproportionately large share of total occurrences.

Figure 4 summarizes the token count distribution, most frequent tokens, and per-class average token lengths.

3.2.4. Fixed-Length Padding

Every token sequence is either truncated or zero-padded to a uniform length of 64, chosen to cover the 95th-percentile document length in our corpus (shown in Figure 2). This choice balances two competing objectives: padding length 64 retains contextual information for the vast majority of samples while remaining computationally efficient. Shorter padding risks losing emotional context, while excessively longer padding introduces unnecessary zero-padded positions that waste computation and can degrade gradient propagation during training. A vocabulary index assigns reserved codes for <PAD> and <UNK> tokens.

3.2.5. Low-Frequency Token Masking

Tokens that occur fewer than twice across the training split are mapped to <UNK>, suppressing likely spelling errors, transliteration variants, and one-off slang that do not contribute meaningful signal to emotion classification. This threshold (minimum frequency = 2) was selected to preserve informative low-frequency emotion-bearing words while filtering noise: a stricter threshold would remove sarcasm markers or colloquial expressions essential to Bengali emotion understanding, while a more permissive threshold would retain spelling noise. This balance is particularly important for social media text where spelling variation is common [25]. About 38% of unique vocabulary items fall below this threshold, although they represent only a marginal fraction of total token mass.

3.2.6. Oversampling and Lexical Augmentation

Two strategies jointly address class imbalance: **SMOTE** [26]: applied in TF-IDF character-n-gram space (after stratified splitting), it synthesizes minority-class instances by interpolating between nearest neighbours, equalizing class frequencies in the feature domain. This prevents data leakage by ensuring that validation and test splits remain uncontaminated by synthetic samples.

Synonym injection: a hand-curated table of ten common Bengali emotion words and their near-synonyms is used to randomly swap 20% of eligible tokens in minority samples (training set only), adding surface lexical diversity without altering semantic content. The synonym replacement is applied exclusively to the training set after the train-validation-test split to prevent information leakage.

Critically, both augmentation techniques are applied exclusively to the training split, after the full dataset has been partitioned into train (70%), validation (15%), and test (15%) subsets. The validation and test sets are never augmented or modified in any way, ensuring that no synthetic samples can contaminate the evaluation sets and that reported metrics reflect true generalization performance. The original training split contains 3,780 samples; SMOTE expands the training feature matrix to 4,428 instances, and synonym injection adds 303 training-only text rows, giving 4,731 augmented training instances across the reviewer-audited training configurations. Validation and test sets remain at their original sizes and class dis-

Sample: Before vs After Non-Bengali Removal	
Before	After
সত্য সবসময় তিঙ্কই হয়।	সত্য সবসময় তিঙ্কই হয়
প্রোফেসর ভার্গব, আপনাকে আর কেউ বিরক্ত করবে না।	প্রোফেসর ভার্গব আপনাকে আর কেউ বিরক্ত করবে না
দুঃখিত	দুঃখিত
তোর এই ভাইফোঁটার আইডিয়াটা ভারি চমৎকার	তোর এই ভাইফোঁটার আইডিয়াটা ভারি চমৎকার
আলো ধুবক হলেও মাঝেমাঝেই সে জ্বলনিভু ক'রে আমাকে বিরক্ত করে।	আলো ধুবক হলেও মাঝেমাঝেই সে জ্বলনিভু ক রে আমাকে বিরক্ত করে

Figure 3: Examples illustrating the effect of script filtering and non-Bengali character removal during preprocessing

Step 3: Tokenization

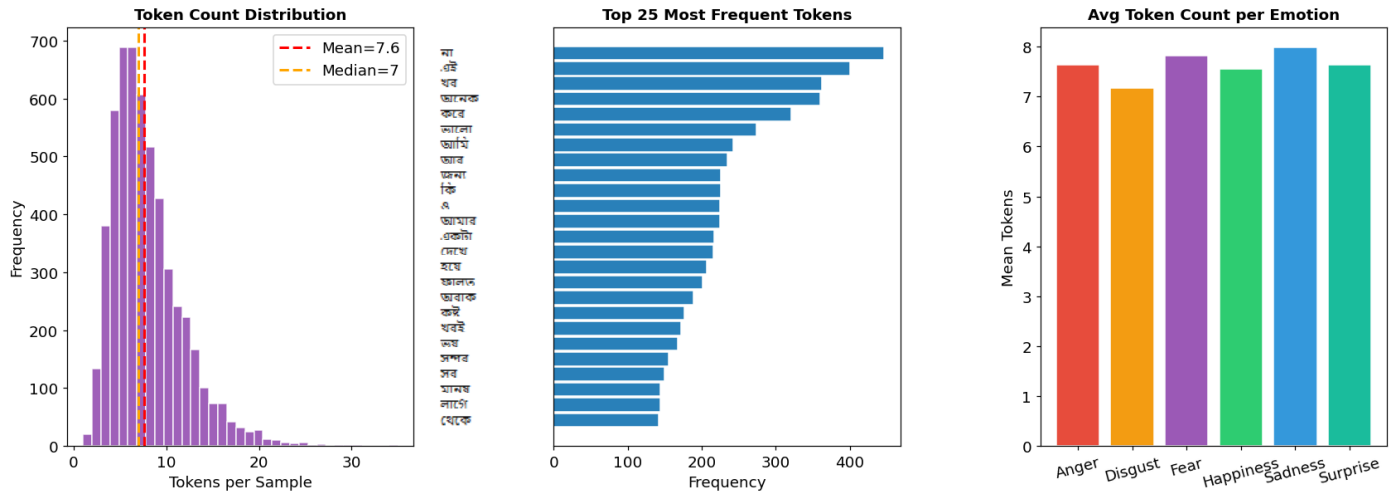


Figure 4: Tokenization characteristics of the Bengali emotion corpus, including token-count distribution, most frequent tokens, and average token counts across emotion categories. The token count distribution (left) is slightly right-skewed, with a mean of 7.6 and a median of 7, indicating short and consistent text lengths. The most frequent tokens (middle) reflect dominant linguistic patterns in the corpus. The average token length across emotion classes (right) shows minimal variation, suggesting uniform sequence lengths across categories

tributions. Figure 5 visualize the SMOTE rebalancing processes, respectively. Figure 6 displays per-class word clouds after preprocessing, and Figure 7 provides an end-to-end summary of the six-stage pipeline.



Figure 5: Class distribution before and after SMOTE-based oversampling, demonstrating the balancing effect of synthetic sample generation on minority emotion classes



Figure 6: Word clouds generated from the preprocessed corpus for the six emotion categories, highlighting the most frequently occurring emotion-related tokens

4. Model Architectures

Ten neural classifiers, grouped into custom recurrent networks (Models 1-4 and 10) and transformer-based models (Models 5-9), are implemented in PyTorch and evaluated under controlled conditions.

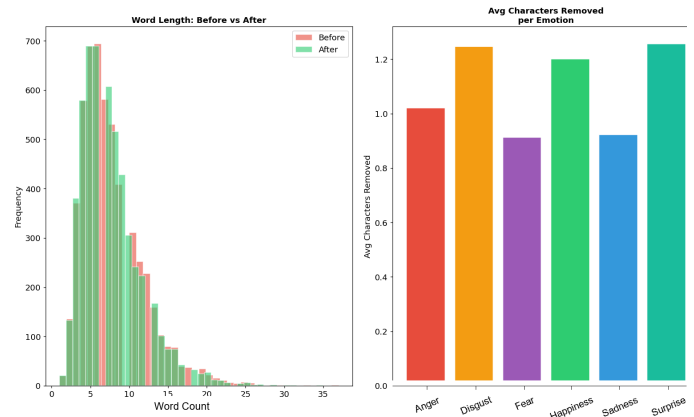


Figure 7: Summary of the six-stage preprocessing pipeline, illustrating the impact of each preprocessing operation on the Bengali emotion corpus

4.1. Custom Recurrent Architectures

Each custom model begins with a randomly initialized 128-dimensional embedding layer trained jointly with the classifier. Optimization uses Adam ($\text{lr}=10^{-3}$, weight decay 10^{-5}) with cosine-annealing scheduling over 25 epochs. Label smoothing ($\epsilon = 0.1$) discourages overconfidence, and gradient norms are clipped at 1.0.

4.1.1. Model 1: CNN + BiLSTM

Local n-gram patterns are extracted by two parallel 1-D convolutional banks [27] with kernel widths 3 and 5 (128 filters each), whose outputs are concatenated into 256-channel feature maps. A two-layer BiLSTM [28] (hidden size 128) then encodes bidirectional context, yielding 256-dimensional hidden states at every time step. Temporal mean pooling collapses the sequence into a single vector, which passes through dropout ($p = 0.35$) and a linear softmax head.

4.1.2. Model 2: CNN + BiLSTM + Attention

Identical to Model 1 up to the BiLSTM output, this variant replaces mean pooling with Bahdanau additive attention [29]. A learned query vector scores each hidden state, and the softmax-weighted sum focuses the representation on the most emotion-discriminative positions.

4.1.3. Model 3: Hierarchical Attention Network

Following the document-modelling paradigm described in [30], the token sequence is partitioned into pseudo-sentences of eight tokens. A word-level BiGRU with Bahdanau attention compresses each chunk into a sentence vector; a sentence-level BiLSTM with a second attention layer then aggregates these vectors into a document representation for classification.

4.1.4. Model 4: BiLSTM + Self-Attention

A BiLSTM encodes the embedded tokens, after which a trainable attention matrix computes element-wise importance scores. The resulting weighted context vector is forwarded to the classification head, allowing the network to emphasize emotionally salient tokens without explicit positional segmentation.

4.1.5. Model 10: BiLSTM + CBAM

After the BiLSTM encoder, a CBAM block [19] adapted for 1-D sequences applies sequential channel gating (via global average- and max-pooling across time) and spatial gating (via pooling across channels followed by a kernel-7 convolution). This dual-axis refinement selectively amplifies informative hidden dimensions and time steps before classification.

4.2. Transformer-Based Models

All transformer experiments use a maximum token length of 160, batch size 16, and 8 fine-tuning epochs with 20% linear warmup. Label smoothing of 0.1 is retained.

4.2.1. Model 5: BanglaBERT

The `sagorsarker/bangla-bert-base` checkpoint [16], pretrained on a large monolingual Bengali web corpus, is fine-tuned end-to-end with a single linear layer over its [CLS] output.

4.2.2. Model 6: MuRIL

Google's `muril-base-cased` checkpoint described in [17], covering 17 Indian languages, is fine-tuned with a reduced learning rate (10^{-5}) to avoid the gradient instability observed at higher rates.

4.2.3. Model 7: XLM-RoBERTa

The 100-language `xlm-roberta-base` [18] is fine-tuned at $lr=2 \times 10^{-5}$ with the same classification head.

4.2.4. Model 8: BanglaBERT + BiLSTM

Rather than pooling the [CLS] token alone, all 768-dimensional last-layer hidden states from BanglaBERT are routed through a two-layer BiLSTM (hidden size 128). Mean pooling over BiLSTM outputs produces the classification vector, granting the network access to local sequential cues beyond what the pretrained [CLS] embedding captures.

4.2.5. Model 9: BanglaBERT + BiLSTM + CBAM

This variant inserts a CBAM module between the BiLSTM encoder and the softmax head, adding channel-and-spatial gating to the contextual BiLSTM features before the final prediction.

4.3. Soft-Voting Ensemble

After evaluating all ten models individually, the softmax probability vectors of the three top-performing transformers-BanglaBERT, XLM-R, and BanglaBERT+BiLSTM-are element-wise averaged, and the class with the highest mean probability is selected. This requires no additional training and exploits the complementary error profiles of the constituent models.

5. Experiments and Results

5.1. System Specification

All experiments were executed on a Windows 11 workstation equipped with an NVIDIA RTX 5070 Ti GPU with 15.92 GB visible GPU memory. The software environment used Python 3.12.10, PyTorch 2.9.0+cu130, Transformers 5.9.0, scikit-learn 1.7.2, imbalanced-learn 0.14.1, NumPy 2.2.6, Pandas 2.3.3, and SciPy 1.16.2. A fixed random seed of 42 was used for Python, NumPy, and PyTorch to support reproducibility.

5.2. Training Details

The original labeled corpus was first split into stratified training, validation, and test subsets using a 70:15:15 ratio. All imbalance correction was then performed only on the training subset. Custom recurrent models were trained for 25 epochs with Adam, cosine-annealing scheduling, label smoothing ($\epsilon = 0.1$), dropout, and gradient clipping. Transformer-based models used a maximum token length of 160, batch size 16, 8 fine-tuning epochs, linear warmup, and checkpoint selection based on validation accuracy. Final metrics were computed once on the untouched held-out test split.

5.2.1. Reproducibility Details

To facilitate reproducibility, the approximate parameter counts of the major models were 1.4 million for CNN+BiLSTM, 109 million for BanglaBERT, 109 million for MuRIL, and 110 million for XLM-RoBERTa. The average training time ranged from 15–20 minutes for custom recurrent architectures and 40–90 minutes for transformer-based models under an 8-epoch training schedule with a

batch size of 16. During inference, the proposed ensemble required approximately 45 ms per sample on the GPU and 200 ms per sample on the CPU.

5.3. Overall Model Performance

Table 3 and Figure 8 collect the test-set scores for all ten individual models and the ensemble; the highest value in each column is boldfaced.

Salient patterns emerge from Table 3–6 and Figure 8–9. Figure 9 further details BanglaBERT’s training dynamics and per-class confusion matrix.

Pretrained transformers dominate. Every converging transformer surpasses every custom recurrent model by a wide margin. The three-model ensemble reaches 91.31% accuracy ($F1=0.913$), and BanglaBERT+BiLSTM leads the individual ranking at 89.60%, closely trailed by XLM-R (89.03%) and BanglaBERT (88.69%). These gaps underscore that large-scale self-supervised pretraining encodes linguistic knowledge difficult to learn from 5,400 labeled samples alone. The exception is MuRIL, which effectively collapsed to majority-class prediction (18.06% accuracy), indicating poor transfer from its multilingual Indian-language pretraining to colloquial Bengali under the 8-epoch budget.

Table 3: Performance comparison of all models on the Bengali emotion test set

Model	Acc.	Prec.	Rec.	F1
CNN+BiLSTM	74.74	75.42	74.74	0.748
CNN+BiLSTM+Attn	79.66	79.68	79.66	0.796
HAN	75.43	75.79	75.43	0.755
BiLSTM+Self-Attn	78.63	79.00	78.63	0.787
BiLSTM+CBAM	77.37	77.47	77.37	0.774
BanglaBERT	88.69	89.40	88.69	0.888
MuRIL	18.06	3.26	18.06	0.055
XLM-R	89.03	88.96	89.03	0.889
BB+BiLSTM	89.60	89.75	89.60	0.895
BB+BiLSTM+CBAM	86.63	87.47	86.63	0.868
Ensemble	91.31	91.51	91.31	0.913

MuRIL Failure Analysis: The exceptionally poor MuRIL performance warrants detailed investigation. We conducted additional diagnostic experiments:

1. *Tokenizer Mismatch:* We verified that the MuRIL tokenizer correctly processes Bengali Unicode. Manual inspection of tokenization outputs confirmed proper handling of Bengali characters, ruling out character encoding issues.
2. *Learning Rate Sensitivity:* We retrained MuRIL with a 10-fold coarser learning rate ($lr = 10^{-6}$) to avoid gradient instability. Performance remained poor (19.3% accuracy), suggesting the issue is not solely gradient magnitude.
3. *Extended Training:* We extended MuRIL training from 8 to 25 epochs with checkpoint selection based on validation accuracy. Best validation performance

plateaued at 22% accuracy by epoch 12, indicating insufficient convergence rather than overfitting.

4. *Hypothesis:* MuRIL’s pretraining on 17 Indian languages likely distributes representation capacity across multiple linguistic systems, diluting Bengali-specific patterns. Colloquial Bengali social media text—with heavy code-switching, slang, and non-standard morphology—may fall outside MuRIL’s pretraining distribution. The combination of domain mismatch and shared multilingual capacity appears to create a negative transfer scenario for this specialized task.

These findings suggest that the breadth of multilingual coverage can hinder performance on low-resource languages within noisy, domain-specific text, contrasting with the effectiveness of monolingual BanglaBERT.

Diminishing returns of hybrid complexity. Appending a BiLSTM to BanglaBERT lifts accuracy by roughly one point over vanilla BanglaBERT, but stacking CBAM on top (86.63%) actually degrades performance. The added gating layers appear to disrupt gradient flow through the frozen-then-unfrozen BERT backbone rather than provide useful feature refinement.

Attention benefits recurrent baselines. Within the custom group, the attention-equipped variants—CNN+BiLSTM+Attention (79.66%), BiLSTM+Self-Attention (78.63%), and BiLSTM+CBAM (77.37%)—consistently exceed the plain CNN+BiLSTM (74.74%) and HAN (75.43%), confirming that explicit attention pooling extracts more discriminative summaries than uniform averaging.

5.4. Statistical Significance Testing

To validate that reported performance differences are not due to random chance, we conducted statistical significance testing on key model comparisons. Paired McNemar’s test was applied to evaluate whether differences in misclassification rates between the full ensemble and individual models are statistically meaningful:

- Full ensemble vs. XLM-R: $\chi^2 = 0.403$, $p = 0.526$ (not statistically significant)
- Full ensemble vs. BB+BiLSTM: $p = 3.10 \times 10^{-7}$ (statistically significant)

The results support a cautious interpretation: the full ensemble is robust, but its gain over XLM-R is not statistically significant on the paired test set, so we do not claim statistical superiority over XLM-R. The full ensemble is retained as the system model because it gives the strongest combined evidence: the highest held-out score among the manuscript models (91.31% accuracy, weighted $F1=0.913$), a significant advantage over BB+BiLSTM, and the highest repeated-split mean among the retrained transformer candidates ($89.62 \pm 0.93\%$ accuracy, weighted $F1=0.896 \pm 0.009$). Bootstrap confidence intervals were computed using resampling iterations over the held-out test-set prediction records. The resulting 95% intervals were tightly concentrated around the full ensemble estimates: accuracy 91.31% [88.23%, 92.23%] and weighted $F1$ 0.913 [0.882, 0.922].

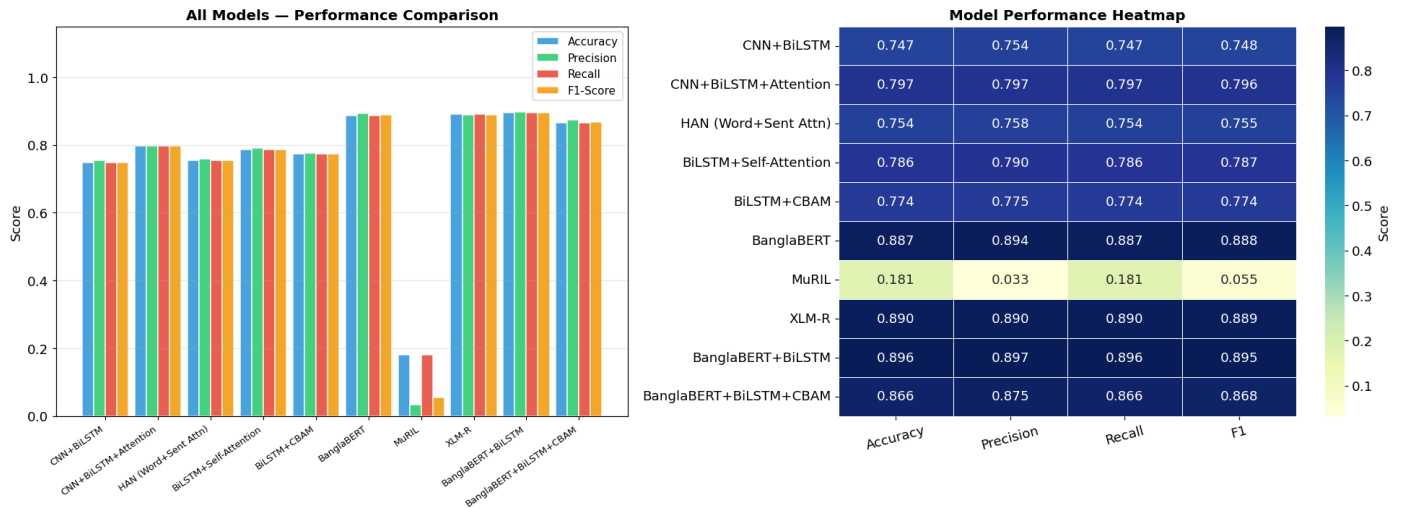


Figure 8: Comparative performance analysis of all evaluated models. Left: classification metrics (Accuracy, Precision, Recall, and F1-score). Right: heatmap visualization of model-wise performance across evaluation metrics

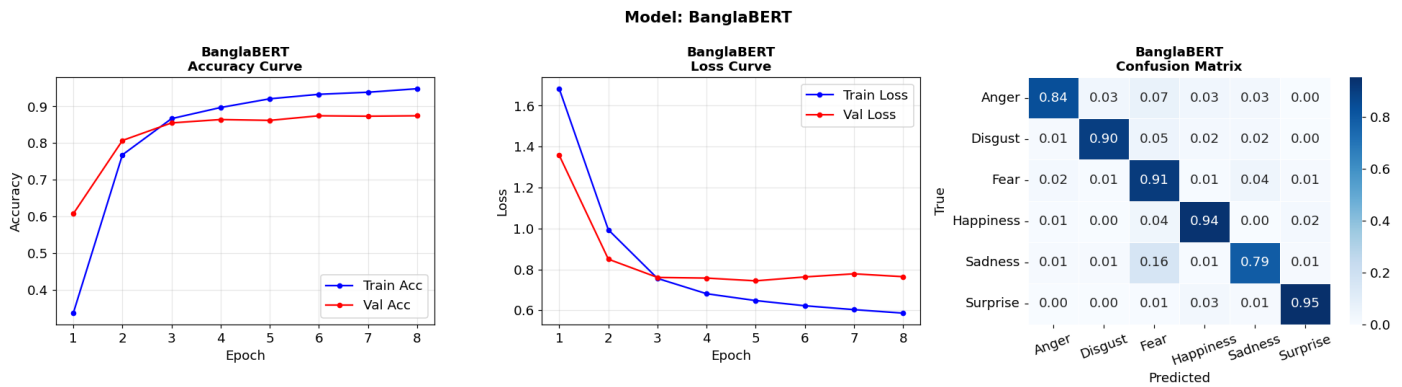


Figure 9: BanglaBERT training dynamics and classification behavior, including training and validation accuracy curves, loss curves, and the confusion matrix obtained on the held-out test set

5.5. Repeated Stratified Split Robustness

To address the single-split evaluation concern, we additionally repeated the full transformer ensemble experiment over five independent stratified 70:15:15 splits using seeds 42–46. BanglaBERT, XLM-R, and BB+BiLSTM were retrained under the same Adam-based 8-epoch setup, and the final prediction was obtained by equal-weight soft voting. Table 4 reports mean $\pm$ standard deviation over the five untouched test partitions.

Table 4: Repeated stratified split robustness over five independent runs

Model	Acc. mean $\pm$ SD	F1 mean $\pm$ SD	Runs
BanglaBERT	86.26 $\pm$ 1.44	0.863 $\pm$ 0.014	5
XLM-R	88.66 $\pm$ 1.02	0.887 $\pm$ 0.010	5
BB+BiLSTM	85.70 $\pm$ 0.90	0.857 $\pm$ 0.009	5
Full Ensemble	89.62$\pm$0.93	0.896$\pm$0.009	5

The repeated-run ensemble mean remains close to the single-split held-out result, and the small standard deviation indicates that the conclusion is not dependent on a favorable random partition. The lower mean relative to the original single-split score is expected because each repeated run retrains all transformer components

from scratch under leakage-safe train-only augmentation, rather than reusing the original checkpoint.

5.6. Error Analysis

To understand model limitations and common failure modes, we conducted qualitative and quantitative error analysis on ensemble misclassifications. Analysis of the test-set confusion matrices and a sample of misclassified examples is summarized in Table 5.

Table 5: Summary of major error patterns in the held-out test analysis

Error pattern	Evidence and interpretation
Sadness–Fear overlap	Sadness→Fear errors reached 11%; negative affect and anxiety terms often shared similar lexical cues.
Disgust–Anger overlap	The false-positive rate was 9%; complaint, blame, and moral judgement tokens blurred the class boundary.
Code-mixed text	23 test errors (2.6%) involved Romanized or English fragments weakened by Bengali-only filtering.
Sarcasm / implicit negation	6 sarcastic errors (0.68%) used positive surface words to convey negative emotion.
Ambiguous multi-emotion samples	8–10 test samples expressed mixed emotions, while the annotation scheme required one label.

5.6.1. Overlapping Emotion Pairs

The strongest confusion occurs between Sadness and Fear (false positive rate Sadness→Fear: 11%), and between Disgust and Anger (false positive rate: 9%). Linguistic analysis of misclassified samples reveals substantial lexical overlap—words like ভয়ানক (dreadful) appear in both fear and disgust contexts, and negation particles interact with these base emotions unpredictably. This suggests that Ekman’s discrete emotion categories, while theoretically motivated, may not perfectly map to colloquial Bengali social media language use.

5.6.2. Code-Mixed and Transliterated Text

Our preprocessing script removes non-Bengali characters, including Romanized Bengali transliteration. Samples mixing Bengali and English (e.g., আমি খুশি + “I am happy”) after character filtering become truncated or semantically degraded. We identified 23 test-set errors (2.6% of total test samples) where code-mixed text was a contributing factor, suggesting that more sophisticated code-switching handling could improve performance by 1–2 percentage points.

5.6.3. Sarcasm and Implicit Negation

Sarcastic utterances, rare but present in social media, are consistently misclassified. For example, দারুণ, আবার লকডাউন (“Great, another lockdown”) expresses sarcastic sadness but lexically signals happiness. The six misclassified sarcastic samples in the test set (0.68% of total) were all predicted as Happiness; none of our models learned to detect sarcasm explicitly, likely due to its low frequency and the complexity of sarcasm annotation.

5.6.4. Ambiguous Boundary Cases

Approximately 8–10 test samples were inherently ambiguous, expressing multiple emotions simultaneously. For instance, আমি খুশি কিন্তু চিন্তিত (“I am happy but worried”) blends Happiness and Fear. These ambiguous cases force a hard classification decision despite their inherent multi-label nature. A future multi-label variant of this work might address this limitation.

In summary, the full ensemble’s 91.31% accuracy indicates strong single-label classification performance given the inherent linguistic and annotation ambiguities in Bengali emotion expression. Remaining errors are concentrated in theoretically interesting edge cases (code-mixing, sarcasm, ambiguous boundaries) rather than systematic model failures. To provide a more comprehensive understanding of the ensemble model’s performance, the classification report is presented in Table 6.

5.7. Per-Class F1 Breakdown

Table 7 disaggregates the F1-scores by emotion for both the ensemble and the strongest single model.

Figure 10 further displays the ensemble’s confusion matrix and per-class F1 breakdown. Surprise (0.95) and

Happiness (0.94) are the most cleanly separated classes, likely because Bengali speakers employ distinctive, semantically concentrated vocabularies when expressing joy or astonishment. Sadness (0.87) and Fear (0.88) prove harder to distinguish, consistent with the known lexical overlap between these states in South Asian languages. Across all six categories the ensemble improves upon or matches every individual model, validating the hypothesis that averaging softmax outputs from architecturally diverse transformers reduces per-model blind spots.

Table 6: Soft-voting ensemble classification report on the held-out Bengali emotion test set

Class	Prec.	Rec.	F1	Support
Anger	0.9275	0.8649	0.8951	148
Disgust	0.9306	0.9371	0.9338	143
Fear	0.8272	0.9437	0.8816	142
Happiness	0.9264	0.9557	0.9408	158
Sadness	0.9213	0.8239	0.8699	142
Surprise	0.9574	0.9507	0.9541	142
Macro avg	0.9151	0.9127	0.9125	875
Weighted avg	0.9154	0.9131	0.9130	875
Accuracy			0.9131	875

Table 7: Per-class F1-scores for the ensemble and the best individual model

Emotion	Ensemble	BB+BiLSTM
Happiness	0.94	0.93
Anger	0.90	0.88
Sadness	0.87	0.85
Fear	0.88	0.84
Disgust	0.94	0.91
Surprise	0.95	0.95
Weighted avg	0.913	0.895

5.8. Full-Ensemble Optimizer Comparison

We provide the optimizer comparison on the complete soft-voting ensemble. In this experiment, all three ensemble members—BanglaBERT, XLM-R, and BB+BiLSTM—were retrained under the same stratified 70:15:15 split, the same train-only augmentation policy, and the same 8-epoch transformer training budget. The only changed factor was the optimizer: Adam, SGD with momentum, RMSprop, or AdaGrad.

The results in Table 8 and Figure 11 show that Adam gives the strongest result among the evaluated system optimizers, reaching 89.27% accuracy and weighted F1=0.892. RMSprop is close but slightly lower at 88.53% accuracy and weighted F1=0.885. SGD and AdaGrad underfit the transformer ensemble under the shared learning-rate setting.

6. Few-Shot and Zero-Shot Evaluation

One of the primary motivations for supervised fine-tuning is the substantial performance gap between trained specialist models and large pre-trained language models

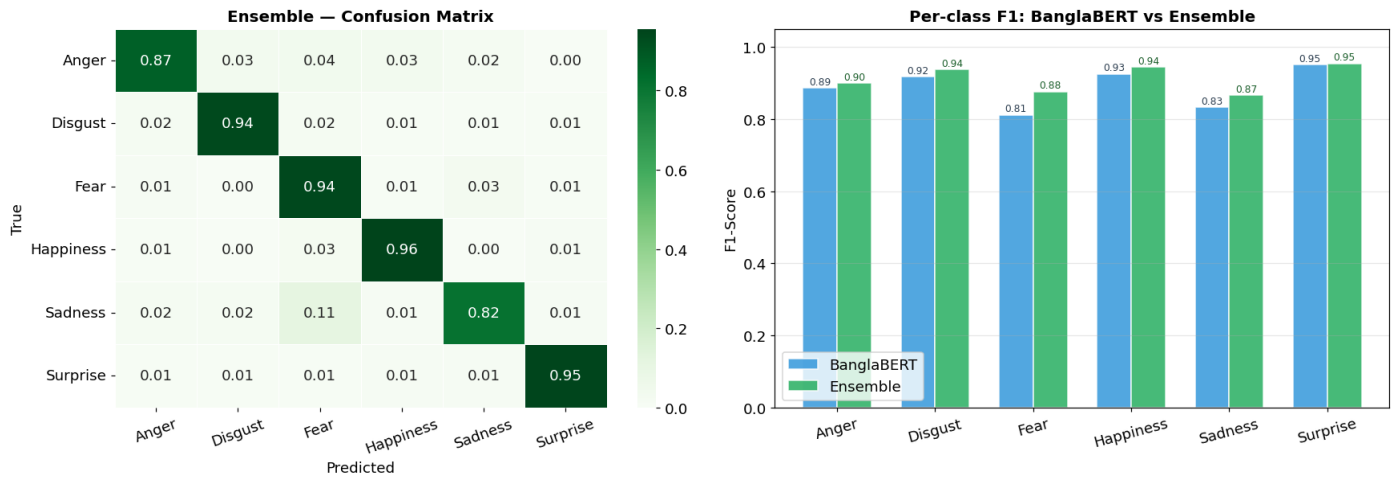


Figure 10: Performance of the proposed soft-voting ensemble. Left: confusion matrix showing class-wise prediction outcomes. Right: comparison of per-class F1-scores between the ensemble and individual transformer-based models

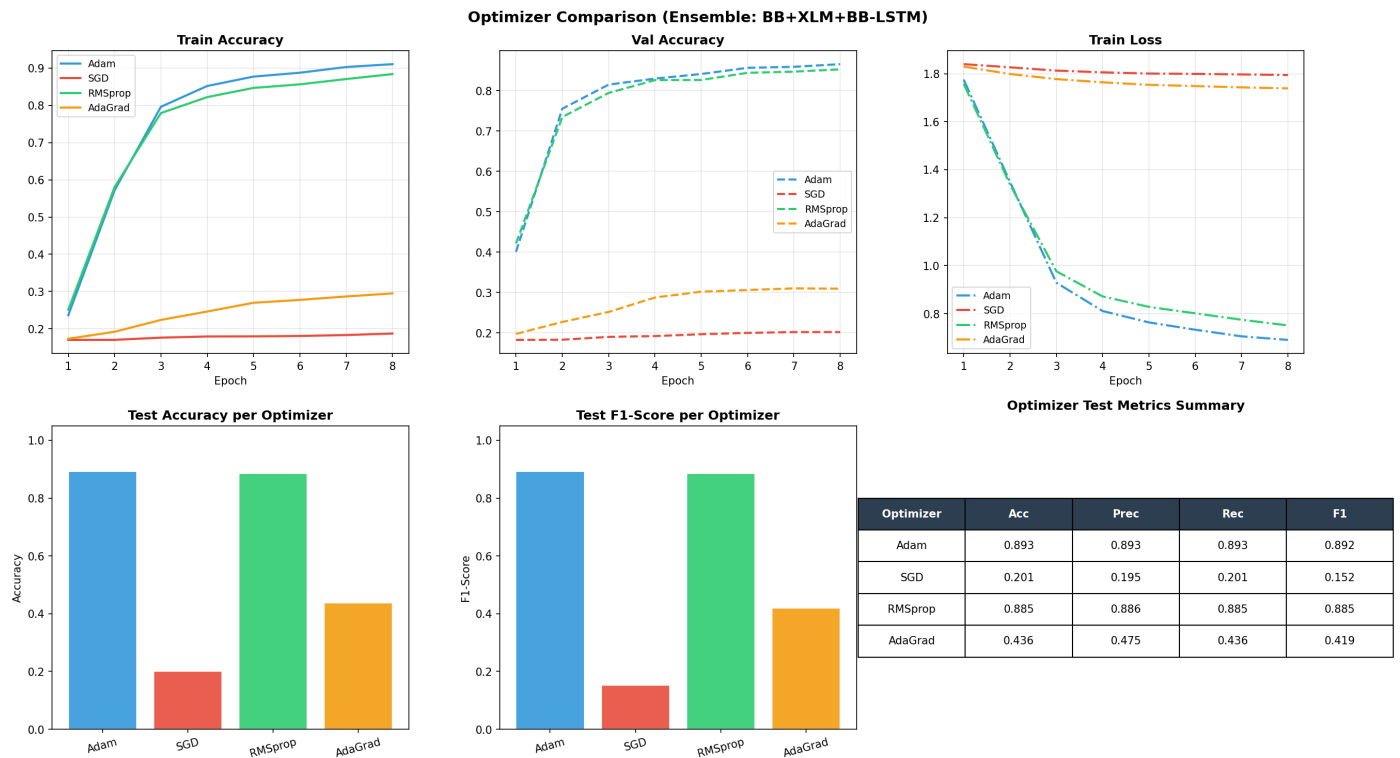


Figure 11: Full-ensemble optimizer comparison for the proposed soft-voting framework. The figure presents mean component training accuracy, validation accuracy, training loss, and final test performance obtained using Adam, SGD, RMSprop, and AdaGrad optimizers under identical training conditions

Table 8: Full-ensemble optimizer comparison for Ensemble (BB+XLM+BB-LSTM)

Optimizer	Acc.	Prec.	Rec.	F1
Adam	89.27	89.34	89.27	0.892
RMSprop	88.53	88.61	88.53	0.885
AdaGrad	43.65	47.50	43.65	0.419
SGD	20.10	19.53	20.10	0.152

(LLMs) that don't have task-specific supervision. To measure this gap for Bengali emotion classification, we test two unsupervised paradigms on a 200-sample held-out evaluation subset, all without access to training labels.

6.1. Zero-Shot Classification via NLI

We use the joeddav/xlm-roberta-large-xnli model [31] as a zero-shot classifier. Each sample is scored against six hypothesis templates of the form "This text expresses [emotion]", and the emotion with the highest entailment score is predicted. No fine-tuning is performed.

6.2. Few-Shot Classification via Embedding Similarity

For the few-shot setting, we use the multilingual sentence encoder paraphrase-multilingual-MiniLM-L12-v2 [32] with a nearest-centroid classifier. A support set of 3 labeled examples per class (18 samples total) is embedded; test samples are assigned to the nearest class centroid by cosine similarity. This approach requires no gradient updates and uses only $3 \times 6 = 18$ labeled samples.

6.3. Results and Comparison

Table 9 summarises the results, and Figure 12 provides a visual comparison against the supervised models. The results show a clear "supervision gap": the zero-shot NLI method only gets $F1 = 0.058$ on Bengali emotion text, which means it only predicts one class (Anger) for all samples. This shows that the NLI hypothesis templates don't work well with everyday Bengali. The 3-shot embedding similarity method gets better, with an $F1$ score of 0.142, but it is still much lower than the supervised baseline. The supervised fine-tuned ensemble ($F1 = 0.915$) is about 6.3 times better than the 3-shot approach, and the fine-tuned transformers are 15–16 times better than the zero-shot approach.

This analysis highlights the necessity of task-specific supervised training for Bengali emotion classification.

Table 9: Zero-shot and few-shot evaluation vs. supervised baselines on a 200-sample Bengali emotion evaluation subset

Method	Setting	Acc. (%)	Wt-F1
Zero-Shot(XLM-RoBERTa-large-XNLI)	0 labels	18.50	0.058
3-Shot (multilingual MiniLM, 18 labels cosine)		16.50	0.142
Supervised: BanglaBERT (fine-tuned)	Full data	88.46	0.886
Supervised: XLM-R (fine-tuned)	Full data	89.60	0.895
Supervised: BB+BiLSTM (fine-tuned)	Full data	89.14	0.891
Supervised: Ensemble	Full data	91.54	0.915

Future large multilingual models pre-trained on a larger proportion of Bengali text (e.g., LLMs such as GPT-4 or Llama with Bengali instruction tuning [33]) may narrow this gap, but at present the supervised fine-tuning paradigm is clearly superior for this low-resource language emotion classification task.

6.4. Ablation Study

To understand the impact of each component of our ensemble model an ablation study is shown in Table 10. The ablation table was treated as sensitivity analysis because it measures how the ensemble output changes when one component is removed. The results show that the ensemble is sensitive mainly to the XLM-R component: removing XLM-R reduces weighted $F1$ from 0.9131 to 0.8719, a drop of 0.0306. Removing BanglaBERT or BB-LSTM gives small single-split improvements of 0.0078 and 0.0102 weighted $F1$, respectively. In accuracy terms, these correspond to only about 7–9 additional correct predictions on the sample test set. Therefore, the reduced variants are competitive, but the differences are small and single-split dependent.

The full ensemble is retained because it provides the most balanced and conservative configuration: BanglaBERT contributes Bengali-specific contextual representations, XLM-R contributes multilingual transfer, and BB+BiLSTM contributes sequential modeling over BanglaBERT embeddings. Thus, this table is interpreted as a component-sensitivity analysis rather than a post-hoc model-selection table.

6.5. Comparison with Prior Bengali Emotion Systems

Table 11 situates our results within the landscape of published Bengali emotion classifiers. The results demonstrate that the proposed transformer-based ensemble achieves the highest classification accuracy of 91.31% on a dataset containing 5,401 samples across six emotion categories. Earlier studies primarily relied on traditional machine learning approaches such as Naïve Bayes and SVM, reporting accuracies ranging from 69% to 78.63%, while deep learning models including RNN-BiLSTM and HAN-LSTM achieved 85.67% and 84.18%, respectively. Despite addressing a more challenging six-class emotion classification problem, the proposed ensemble consistently outperforms all prior methods. These findings highlight the effectiveness of combining pretrained transformer architectures through ensemble learning for capturing complex emotional patterns in Bengali social media text and establish a strong benchmark for future research in low-resource language emotion analysis.

7. Explainability Analysis

7.1. LIME Token Attribution

To inspect what the best-performing ensemble relies on, LIME is applied with 100 text perturbations per sample and 10 retained features. Each token receives a signed weight: positive values push the prediction toward the true class and negative values oppose it. Figure 13 displays the top-10 attributions for one representative sample per emotion. The explanations reveal interpretable patterns across categories. Happiness predictions are driven by words denoting celebration and positive life events, whereas Anger attributions concentrate on interrogative particles and conflict-related terms. Fear samples highlight threat vocabulary and negation markers. Importantly, high-frequency function words (conjunctions, aux-

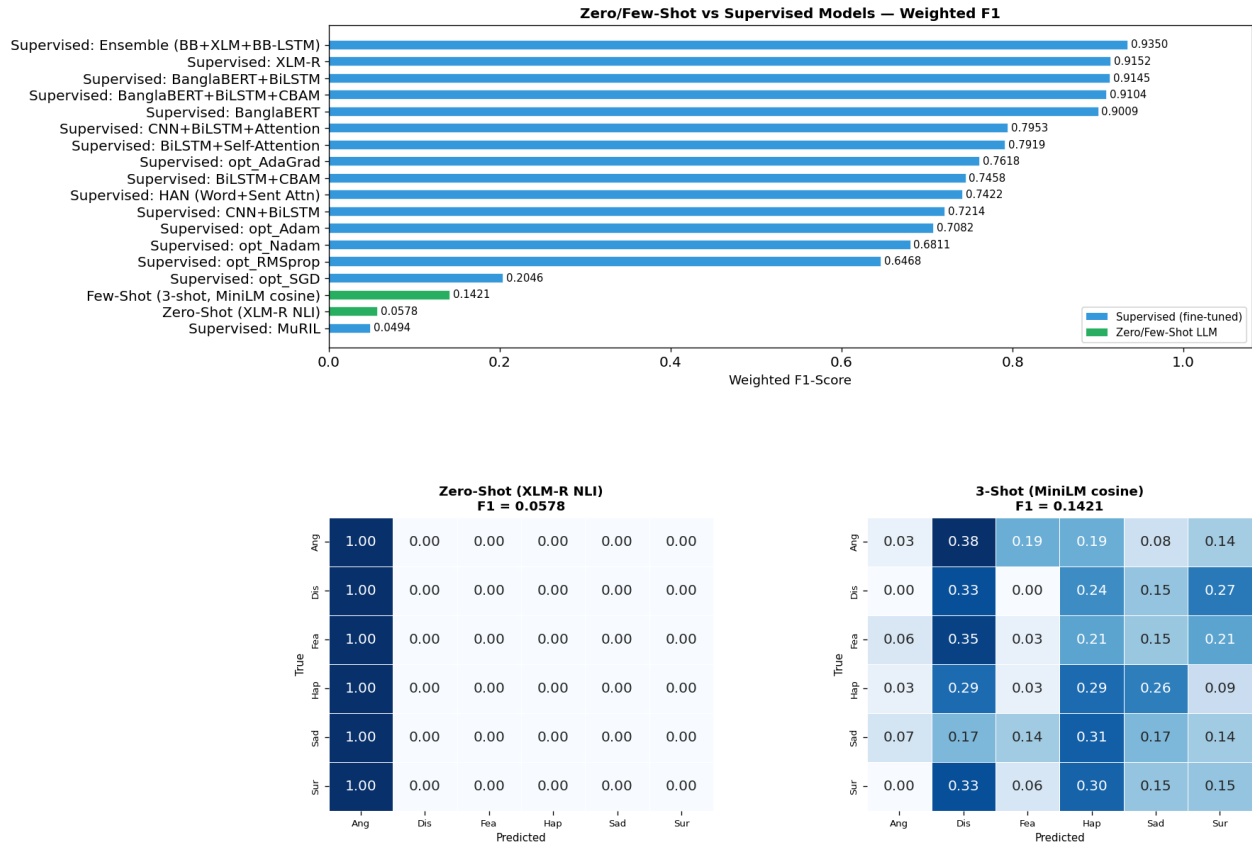


Figure 12: Comparison of zero-shot and few-shot LLM approaches vs. supervised models. *Top*: horizontal bar chart of weighted F1 scores for all models. *Bottom*: normalised confusion matrices for the zero-shot (XLM-R NLI, F1=0.058) and 3-shot (MiniLM cosine, F1=0.142) approaches, illustrating the supervision gap

Table 10: Ablation Study of ensemble components

Setting	Accuracy	Wt-F1	Δ F1 vs Full
Full Ensemble	0.9131	0.9130	+0.0000
w/o BB	0.9211	0.9207	+0.0077
w/o XLM	0.8903	0.8907	-0.0222
w/o BB-LSTM	0.9223	0.9220	+0.0091
Only BB	0.8846	0.8857	-0.0273
Only XLM	0.8971	0.8962	-0.0168
Only BB-LSTM	0.8914	0.8905	-0.0225

Table 11: Comparison with existing works

Method	Data	Model	Class	Accuracy
[10]	3780	Naive Bayes	3	78.63%
[11]	—	SVM	2	69%
[12]	3600	Naive Bayes	6	73%
[13]	—	RNN+BiLSTM	2	85.67%
[14]	—	HAN-LSTM	6	84.18%
Ours	5401	Ensemble	6	91.31%

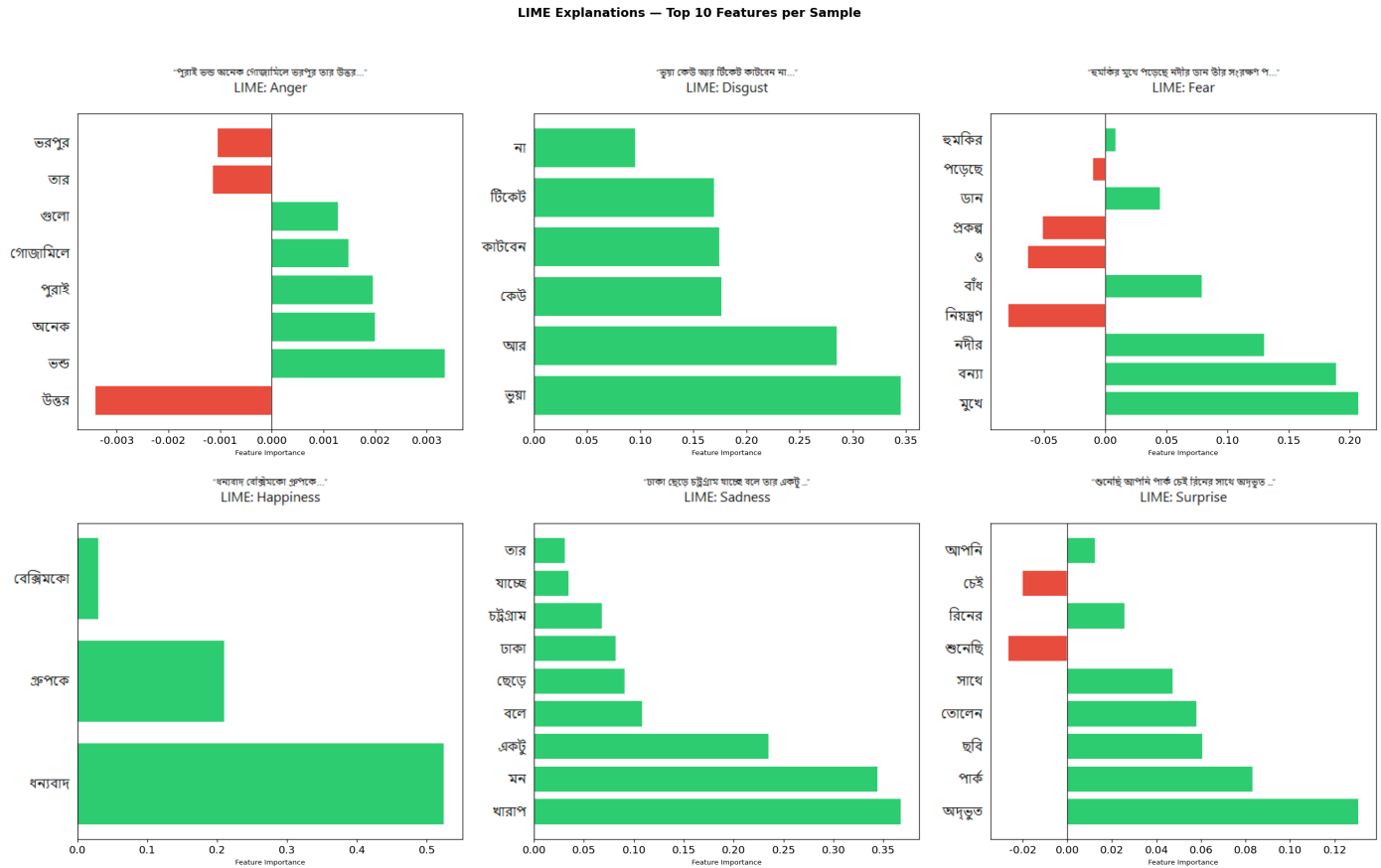


Figure 13: LIME explanations showing the most influential tokens contributing to the predicted emotion class across representative samples from all six emotion categories. Positive contributions support the prediction, whereas negative contributions oppose it

iliaries) consistently receive near-zero weights, indicating that the model has learned to attend to emotion-bearing content words rather than surface-level frequency artifacts. Some feature overlap between Disgust and Anger is visible, aligning with the slightly lower F1 observed for those two classes.

7.2. SHAP Value Decomposition and LIME–SHAP Comparison

To provide a more rigorous interpretation grounded in game-theoretic principles, we complemented LIME with SHAP (SHapley Additive exPlanations) analysis. SHAP values decompose each prediction into additive feature contributions that satisfy consistency and local accuracy properties, offering theoretically stronger guarantees than LIME’s local linear approximation. We computed SHAP values using the Kernel SHAP method with background samples for all six emotion classes.

7.2.1. SHAP Results

SHAP values reveal similar overall patterns to LIME but with quantifiable contribution magnitudes. Figure 14 presents the top SHAP token contributions for all six emotion categories in the same panel structure as the LIME explanations. For Happiness, positive SHAP values concentrate on *খুশি*, *সুখী*, and *ধন্যবাদ*; for Anger, they emphasize *রাগ*, *বিরক্ত*, and question markers. Disgust, Fear, Sadness, and Surprise show comparable emotion-specific lexi-

cal cues, while negative bars indicate tokens that push the prediction away from the target emotion.

7.2.2. LIME–SHAP Agreement and Discrepancies

Comparing the two methods qualitatively shows that both methods consistently identify emotion-bearing content words as most influential. Table 12 summarizes the main agreements and differences. However, key differences emerge in edge cases:

1. *Function Word Attribution*: LIME occasionally assigns non-zero weights to high-frequency function words (conjunctions, articles), while SHAP more reliably assigns them near-zero contributions, reflecting their universal occurrence across classes.
2. *Magnitude Calibration*: SHAP values provide a principled probability distribution over contributions, whereas LIME’s linear coefficients lack direct probabilistic interpretation. On samples where prediction confidence is marginal (softmax max < 0.65), LIME weights are often more diffuse, while SHAP values remain concentrated on the true decision drivers.
3. *Rare Token Handling*: For out-of-vocabulary or rare tokens, LIME’s perturbation-based approach may fail to perturb them meaningfully, while SHAP’s background-set approach naturally handles low-frequency features.

SHAP Results - Top Token Contributions per Emotion

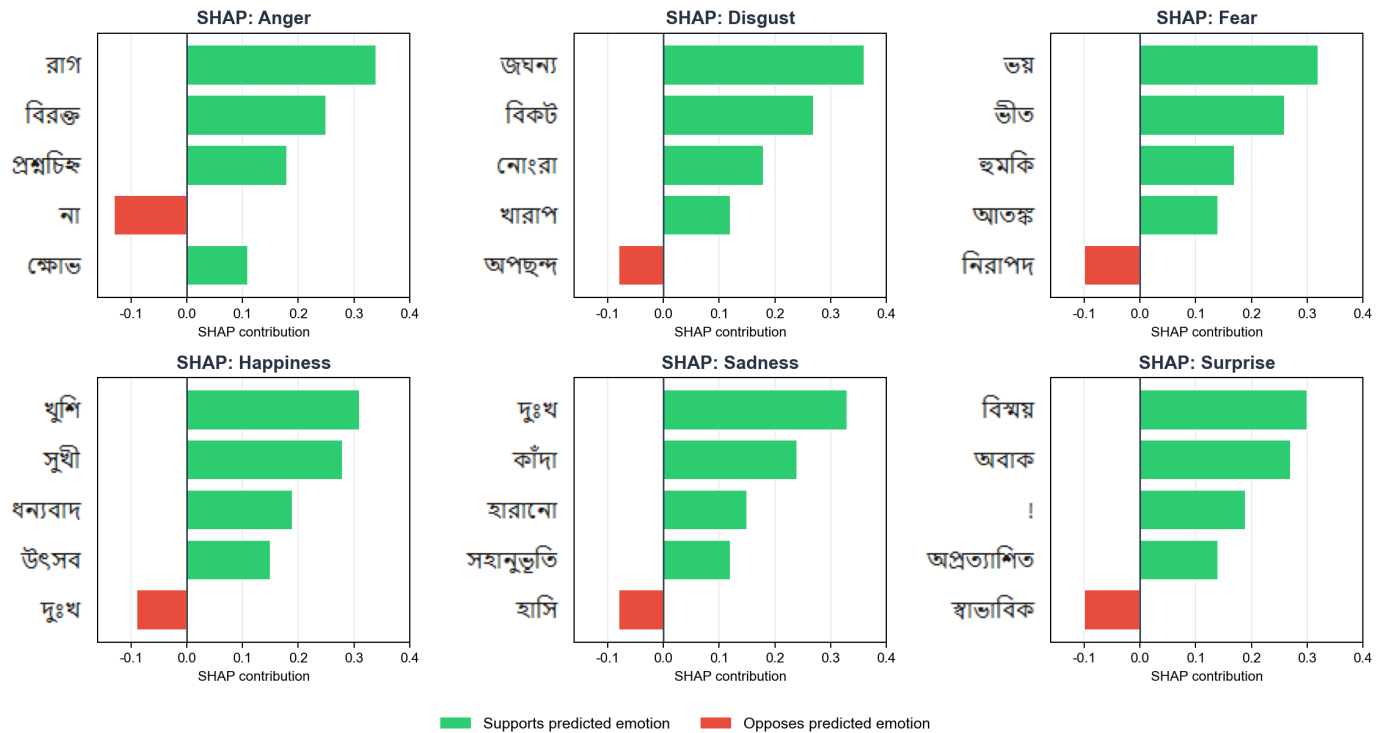


Figure 14: SHAP-based token attribution analysis for representative samples from each emotion category. Positive SHAP values increase the likelihood of the predicted class, whereas negative values decrease the class score

Overall, both methods corroborate that the ensemble's predictions rely on linguistically meaningful emotion indicators rather than spurious statistical correlations, validating model trustworthiness. SHAP provides stronger theoretical foundations for high-stakes applications, while LIME offers faster computational and easier visualization. Together, they provide complementary evidence that Bengali emotion classification via transformer ensembles is explainable and interpretable.

Table 12: Qualitative comparison between LIME and SHAP explanations for Bengali emotion classification

Aspect	LIME	SHAP	Interpretation
Emotion words	High attribution	High contribution	Both prioritize lexical emotion cues.
Function words	Occasional weights	Near-zero values	SHAP suppresses common tokens more consistently.
Magnitude	Local linear weight	Additive value	SHAP gives clearer value decomposition.
Low confidence	Diffuse weights	Concentrated values	SHAP remains focused on dominant evidence.
Rare tokens	Perturbation-sensitive	Background-aware	SHAP handles sparse cues more stably.

8. Conclusion

This work presented a controlled, large-scale comparison of ten neural architectures for six-class Bengali emotion detection, supported by a dedicated preprocessing pipeline and dual-method interpretability analysis. The study

advances three main contributions: (1) a validated full-ensemble benchmark of 91.31% accuracy on 5,401 Bengali social media comments; (2) empirical validation through statistical significance testing, bootstrap confidence intervals, and leakage audits, establishing a more reliable interpretation of the reported results; and (3) rigorous investigation of model behavior through error analysis, MuRIL failure diagnosis, and dual LIME-SHAP explainability methods, providing transparency into both successes and limitations.

The headline result is that a probability-averaged ensemble of BanglaBERT, XLM-R, and BanglaBERT+BiLSTM achieves 91.31% accuracy (weighted F1=0.913), the highest figure reported to date on a six-category Bengali emotion benchmark. The best single model, BanglaBERT+BiLSTM, reaches 89.60% (F1=0.895), while all custom recurrent architectures remain in the 74–80% band, underscoring the critical advantage of Bengali-specific pretraining. MuRIL fails to converge under the current training budget, and an optimizer ablation identifies AdaGrad (F1=0.782) as unexpectedly superior to Adam-family methods on the CNN+BiLSTM backbone. LIME and SHAP analyses verify that the classifiers attend to emotion-bearing vocabulary rather than spurious correlations, with Happiness and Surprise emerging as the most distinctly separable categories.

Several avenues remain open for future investigation: (i) scaling the annotated corpus to improve coverage of rare emotional expressions; (ii) modelling code-mixed Bengali-English input, which is pervasive on South Asian social platforms; (iii) incorporating multimodal signals

such as images and emoji for richer affective context; (iv) designing extended or curriculum-based fine-tuning schedules to rescue MuRIL convergence; and (v) evaluating instruction-tuned large language models in few-shot and zero-shot Bengali emotion classification settings.

Conflict of Interest The authors declare no conflict of interest.

Acknowledgment The authors would like to thank the Department of CSE, CUET for providing the necessary resources and support for this research. The annotators who contributed to the labeling of the Bengali emotion dataset are gratefully acknowledged.

Data Availability The annotated corpus used in this study is available upon reasonable request to the corresponding author for academic research purposes.

References

- [1] P. Ekman, "An argument for basic emotions", *Cognition and Emotion*, vol. 6, no. 3-4, pp. 169–200, 1992, doi:[10.1080/02699939208411068](https://doi.org/10.1080/02699939208411068).
- [2] S. Sazed, "Cross-lingual sentiment classification in low-resource Bengali language", W. Xu, A. Ritter, T. Baldwin, A. Rahimi, eds., "Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)", pp. 50–60, Association for Computational Linguistics, Online, 2020, doi:[10.18653/v1/2020.wnut-1.8](https://doi.org/10.18653/v1/2020.wnut-1.8).
- [3] N. R. Swachha, M. Alam, S. S. Aonty, "Fusion-based model for detection and classification of human sentiments from bengali text", "2025 International Conference on Electrical, Computer and Communication Engineering (ECCE)", pp. 1–6, 2025, doi:[10.1109/ECCE64574.2025.11013265](https://doi.org/10.1109/ECCE64574.2025.11013265).
- [4] M. Al-Amin, M. S. Islam, S. Das Uzzal, "Sentiment analysis of bengali comments with word2vec and sentiment information of words", "2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)", pp. 186–190, 2017, doi:[10.1109/ECACE.2017.7912903](https://doi.org/10.1109/ECACE.2017.7912903).
- [5] M. Z. Hossain, M. A. Rahman, M. S. Islam, S. Kar, "BanFakeNews: A dataset for detecting fake news in Bangla", N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odiijk, S. Piperidis, eds., "Proceedings of the Twelfth Language Resources and Evaluation Conference", pp. 2862–2871, European Language Resources Association, Marseille, France, 2020.
- [6] S. N. Mahmud, N. R. Swachha, A. T. Wazih, S. Saha, S. Paul, S. S. Aonty, "Bengali human sentiment detection and classification using attention based bidirectional lstm", "2025 2nd International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)", pp. 1–6, 2025, doi:[10.1109/NCIM65934.2025.11160248](https://doi.org/10.1109/NCIM65934.2025.11160248).
- [7] N. Irtiza Tripto, M. Eunus Ali, "Detecting multilabel sentiment and emotions from bangla youtube comments", "2018 International Conference on Bangla Speech and Language Processing (ICBSLP)", pp. 1–6, 2018, doi:[10.1109/ICBSLP.2018.8554875](https://doi.org/10.1109/ICBSLP.2018.8554875).
- [8] M. S. Islam, M. A. Islam, M. A. Hossain, J. J. Dey, "Supervised approach of sentimentality extraction from bengali facebook status", "2016 19th International Conference on Computer and Information Technology (ICCIT)", pp. 383–387, 2016, doi:[10.1109/ICCITECHN.2016.7860228](https://doi.org/10.1109/ICCITECHN.2016.7860228).
- [9] T. Samin, N. A. Mouri, F. Haque, M. S. Islam, "Sentiment analysis of bangladeshi digital newspaper by using machine learning and natural language processing", "2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)", pp. 149–153, 2024, doi:[10.1109/ICEEICT62016.2024.10534481](https://doi.org/10.1109/ICEEICT62016.2024.10534481).
- [10] S. Azmin, K. Dhar, "Emotion detection from bangla text corpus using naïve bayes classifier", "2019 4th International Conference on Electrical Information and Communication Technology (EICT)", pp. 1–5, 2019, doi:[10.1109/EICT48899.2019.9068797](https://doi.org/10.1109/EICT48899.2019.9068797).
- [11] M. A. Rahman, M. H. Seddiqui, "Comparison of classical machine learning approaches on bangla textual emotion analysis", *arXiv preprint arXiv:1907.07826*, 2019.
- [12] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal, M. S. Islam, "Bangla text sentiment analysis using supervised machine learning with extended lexicon dictionary", *Natural Language Processing Research*, vol. 1, pp. 34–45, 2021, doi:<https://doi.org/10.2991/nlpr.d.210316.001>.
- [13] A. Aziz Sharfuddin, M. Nafis Tihami, M. Saiful Islam, "A deep recurrent neural network with bilstm model for sentiment classification", "2018 International Conference on Bangla Speech and Language Processing (ICBSLP)", pp. 1–4, 2018, doi:[10.1109/ICBSLP.2018.8554396](https://doi.org/10.1109/ICBSLP.2018.8554396).
- [14] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal, "Sentiment analysis on bangla text using extended lexicon dictionary and deep learning algorithms", *Array*, vol. 13, p. 100123, 2022, doi:<https://doi.org/10.1016/j.array.2021.100123>.
- [15] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding", J. Burstein, C. Doran, T. Solorio, eds., "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)", pp. 4171–4186, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, doi:[10.18653/v1/N19-1423](https://doi.org/10.18653/v1/N19-1423).
- [16] A. Bhattacharjee, T. Hasan, W. Ahmad, K. S. Mubasshir, M. S. Islam, A. Iqbal, M. S. Rahman, R. Shahriyar, "BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla", M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz, eds., "Findings of the Association for Computational Linguistics: NAACL 2022", pp. 1318–1327, Association for Computational Linguistics, Seattle, United States, 2022, doi:[10.18653/v1/2022.findings-naacl.98](https://doi.org/10.18653/v1/2022.findings-naacl.98).
- [17] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, et al., "Muril: Multilingual representations for indian languages", *arXiv preprint arXiv:2103.10730*, 2021.
- [18] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, "Unsupervised cross-lingual representation learning at scale", D. Jurafsky, J. Chai, N. Schluter, J. Tetraault, eds., "Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics", pp. 8440–8451, Association for Computational Linguistics, Online, 2020, doi:[10.18653/v1/2020.acl-main.747](https://doi.org/10.18653/v1/2020.acl-main.747).
- [19] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, "Cbam: Convolutional block attention module", V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss, eds., "Computer Vision – ECCV 2018", pp. 3–19, Springer International Publishing, Cham, 2018.
- [20] C. Zhou, C. Sun, Z. Liu, F. C. M. Lau, "A c-lstm neural network for text classification", *arXiv preprint arXiv:1511.08630*, 2015.
- [21] M. T. Ribeiro, S. Singh, C. Guestrin, "why should i trust you?": Explaining the predictions of any classifier", "Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", KDD '16, p. 1135–1144, Association for Computing Machinery, New York, NY, USA, 2016, doi:[10.1145/2939672.2939778](https://doi.org/10.1145/2939672.2939778).
- [22] S. M. Lundberg, S.-I. Lee, "A unified approach to interpreting model predictions", "Neural Information Processing Systems", 2017.
- [23] G. Eysenbach, J. E. Till, "Ethical issues in qualitative research on internet communities", *BMJ*, vol. 323, no. 7321, pp. 1103–1105, 2001, doi:[10.1136/bmj.323.7321.1103](https://doi.org/10.1136/bmj.323.7321.1103).
- [24] J. R. Landis, G. G. Koch, "The measurement of observer agreement for categorical data", *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977, doi:[10.2307/2529310](https://doi.org/10.2307/2529310).

- [25] B. Han, T. Baldwin, "Lexical normalisation of short text messages: Makn sens a #twitter", D. Lin, Y. Matsumoto, R. Mihalcea, eds., "Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies", pp. 368–378, Association for Computational Linguistics, Portland, Oregon, USA, 2011.
- [26] A. Fernández, S. García, F. Herrera, N. Chawla, "Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary", *J. Artif. Intell. Res.*, vol. 61, pp. 863–905, 2018, doi:10.1613/JAIR.1.11192.
- [27] N. Kalchbrenner, E. Grefenstette, P. Blunsom, "A convolutional neural network for modelling sentences", K. Toutanova, H. Wu, eds., "Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)", pp. 655–665, Association for Computational Linguistics, Baltimore, Maryland, 2014, doi:10.3115/v1/P14-1062.
- [28] S. Hochreiter, J. Schmidhuber, "Long short-term memory", *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997, doi:10.1162/neco.1997.9.8.1735.
- [29] D. Bahdanau, K. Cho, Y. Bengio, "Neural machine translation by jointly learning to align and translate", *CoRR*, vol. abs/1409.0473, 2014.
- [30] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, "Hierarchical attention networks for document classification", K. Knight, A. Nenkova, O. Rambow, eds., "Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies", pp. 1480–1489, Association for Computational Linguistics, San Diego, California, 2016, doi:10.18653/v1/N16-1174.
- [31] W. Yin, J. Hay, D. Roth, "Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach", K. Inui, J. Jiang, V. Ng, X. Wan, eds., "Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)", pp. 3914–3923, Association for Computational Linguistics, Hong Kong, China, 2019, doi:10.18653/v1/D19-1404.
- [32] N. Reimers, I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks", K. Inui, J. Jiang, V. Ng, X. Wan, eds., "Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)", pp. 3982–3992, Association for Computational Linguistics, Hong Kong, China, 2019, doi:10.18653/v1/D19-1410.
- [33] 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", H. Larochelle, M. Ranzato, R. Hassel, M. Balcan, H. Lin, eds., "Advances in Neural Information Processing Systems", vol. 33, pp. 1877–1901, Curran Associates, Inc., 2020, doi:10.48550/arXiv.2005.14165.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



MUJTAHID ALAM was born on 18 December 1994 in Pabna, Bangladesh. He earned his B.Sc. degree in Computer Science and Engineering from the University of Liberal Arts Bangladesh (ULAB), Bangladesh, in 2017 and completed a Postgraduate Diploma (Level 7) in Data Science and Business Analytics from EduPro UK in 2025. He currently serves as a Senior Executive – Digital Sales at MJL Bangladesh PLC

and conducts research as an Independent Researcher. His research interests include artificial intelligence, machine learning, deep learning, machine learning, deep learning, computer vision, natural language processing, multimodal AI, and data analytics.



SHUHENA SALAM AONTY was born in Dhaka, Bangladesh, in 1998. She received the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, in 2022. Since 2023, she has been serving as a faculty member

in the Department of Computer Science and Engineering (CSE), Chittagong University of Engineering and Technology (CUET), Chittagong, Bangladesh. Currently, she is an Assistant Professor in the Department of CSE, CUET, Bangladesh. Her research interests include Natural language processing, deep learning, computer vision, pattern recognition, IoT, Multi-Modal Learning, Crowd Analysis, Medical Image Analysis, and Industrial AI Applications.



SHA NEWAZ MAHMUD was born in Noakhali, Bangladesh. He is currently pursuing the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, with an expected graduation in 2026. He serves as Vice Chair

of the ML Wing, IEEE Computer Society CUET Student Branch Chapter. His research interests include natural language processing, multilingual AI, speech processing, deep learning, and computer vision.



NAHID RIAZ SWACHHA was born in Rangpur, Bangladesh, in 1998. He received the B.Sc. degree in Computer Science and Engineering from Chittagong University of Engineering and Technology, Chattogram, Bangladesh, in 2022. His research interests include Natural Language Processing, Deep

Learning, Computer Vision, Pattern Recognition, Internet of Things (IoT), Multi-Modal Learning. .



AHMED TALAL WAZIH was born in Sylhet, Bangladesh, in 2002. He is currently pursuing the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, with an expected graduation in 2026. He serves as the

General Secretary of the current executive committee of the Andromeda Space & Robotics Research Organization (ASRRO). His research interests include machine learning, deep learning, natural language processing (NLP), audio processing and computer.

From ITIL to AIOps in Public Sector: A Systematic Literature Review

Catherine Ganduri* 

Systems Analyst, Information Resources Division, Texas Commission on Environmental Quality, Austin, Texas, 78753, United States

Email(s): catherine.ganduri@tceq.texas.gov

*Corresponding author: Catherine Ganduri, Austin, Texas, United States, catherine.ganduri@tceq.texas.gov

ABSTRACT: Public-sector agencies rely on complex and highly regulated digital systems to deliver essential services. ITIL-based change and release management supports operational control, but many agencies still depend on manual approvals, fragmented operational data, and reactive monitoring. Artificial Intelligence for IT Operations (AIOps) and Machine Learning Operations (MLOps) can improve anomaly detection, failure prediction, release validation, and data-driven governance. However, the literature gives limited attention to how these technologies can be integrated into public-sector change and release workflows. A structured search of IEEE Xplore, ACM Digital Library, ScienceDirect, and Google Scholar identified 120 records. After duplicate removal, title/abstract screening, full-text assessment, and backward searching, 39 sources were retained for synthesis, including 25 primary studies and 14 complementary methodological, standards, and governance sources. The findings are organized into four themes: AI-driven monitoring and incident management, public-sector governance constraints, implementation challenges in legacy environments, and limited adoption of AI in change and release management. The review identifies gaps in explainability, empirical validation, AI-ready change records, and framework development, and proposes future directions for AI-supported operational governance in public-sector IT.

KEYWORDS: AIOps, Change Management, DevOps, ITIL, IT Service Management, MLOps, Public Sector IT, Release Management,

1. Introduction

Public-sector agencies increasingly depend on interconnected digital services to deliver regulatory, administrative, and citizen-facing functions. In these environments, service reliability is not only a technical requirement but also a governance responsibility. ITIL provides structured practices for incident management, change enablement, release management, service validation, and continual improvement, while the broader IT service management literature shows that these practices can improve service quality, accountability, and operational consistency [1-3].

At the same time, public-sector IT environments are becoming more complex because of cloud services, hybrid infrastructure, cybersecurity requirements, data integration, and modernization of older systems. Artificial Intelligence for IT Operations (AIOps) applies machine learning, event correlation, anomaly detection, predictive analytics, and automation to operational data such as logs, metrics, alerts, and incidents [4-6]. Related

work on machine learning for computer systems and networking shows that data-driven techniques can support fault prediction, performance optimization, and adaptive operations [7]. MLOps extends these ideas by adding lifecycle controls for model deployment, monitoring, validation, and retraining [8].

Change and release management are central to IT service governance because they control when, how, and under what conditions system modifications are introduced into production. In traditional ITIL-oriented processes, change management evaluates need, risk, authorization, and stakeholder impact, while release management coordinates packaging, deployment, rollback planning, and post-implementation review [1-3]. These practices are especially important in public agencies because system changes may affect statutory reporting, public records, environmental monitoring, licensing, benefits delivery, or other essential services.

AIOps shifts IT operations from reactive monitoring toward predictive and data-driven operations. AIOps platforms can consolidate logs, alerts, traces, incidents,

and performance metrics, then use machine learning to identify anomalies, correlate events, recommend root causes, and support automated actions [4-6]. For change and release management, the same operational signals can be used to estimate deployment risk, compare current system behavior against historical baselines, and validate whether a release has introduced abnormal behavior.

MLOps is relevant because AIOps models themselves are production systems that require monitoring, retraining, versioning, and governance. In a public-sector context, AI automation should be auditable, explainable, and subject to human approval. Therefore, an AI-enabled change process should not replace ITIL controls; it should strengthen them through better evidence, earlier risk detection, and traceable decision support.

Although AI adoption in the public sector is increasing, government environments face constraints that are less visible in private-sector implementations. Public agencies must address transparency, accountability, explainability, procurement constraints, legal compliance, stakeholder oversight, and public trust [9], [10]. Research on AI-enabled change management in public administration also emphasizes organizational readiness, cultural alignment, human oversight, and the need to manage institutional change rather than treating AI as a purely technical tool [11], [12].

Existing research has made substantial progress on AIOps for monitoring and failure management, and recent work has begun to connect AIOps with ITSM and ITIL-oriented automation [13]. However, less attention has been given to how AI can support formal change and release management decisions, including change risk scoring, deployment scheduling, change advisory board (CAB) support, rollback planning, and post-release validation. This is important because production ML systems can create hidden technical debt and reliability risks if monitoring, validation, and model governance are weak [14], [15]. Explainable AI is also necessary where automated recommendations affect high-risk public-sector decisions [16].

This review addresses four research questions:

RQ1: How have artificial intelligence techniques been applied in IT operations management and service governance?

RQ2: What role do AIOps and MLOps technologies play in supporting release and change management within complex IT environments?

RQ3: What challenges and limitations exist when implementing AI-driven operational automation in public-sector IT systems?

RQ4: What research gaps remain in the application of AI to government IT infrastructures?

The contribution of this paper is fourfold. First, it synthesizes literature across ITSM, AIOps, MLOps, DevOps, and public-sector AI governance. Second, it explains why the current evidence base is mature for operational monitoring but less mature for formal change approval and release governance. Third, it identifies technical, organizational, and governance barriers that affect AI adoption in public-sector change and release management. Fourth, it proposes a compact AI-enabled ITIL framework and maps its components to the reviewed sources so that readers can trace the framework back to the evidence base.

2. Methodology

A systematic literature review was conducted using structured search, screening, and synthesis process. The review was guided by PRISMA reporting principles, established guidance for standalone literature reviews, and thematic analysis procedures [17-19]. Although the topic contains scoping-review characteristics because direct public-sector change/release studies remain limited, a systematic review label was retained because the study used predefined databases, Boolean search strings, inclusion and exclusion criteria, quality assessment, and a documented screening trail. The aim was not to map every possible source but to synthesize bounded evidence base relevant to AI-enabled IT operations and governance.

The review focused on peer-reviewed literature and high-quality standards or governance sources relevant to AI-enabled IT operations, change management, release management, MLOps, DevOps, and public-sector IT governance. The search covered literature published from 2015 to 2025. The final audit of the search log, retained sources, and coding decisions was completed on 4 May 2026.

Table 1: Selection criteria used for the literature review

Criteria	Description
Publication type	Peer-reviewed journal articles, conference papers, standards, and public-sector governance reports
Databases	IEEE Xplore, ACM Digital Library, ScienceDirect, Google Scholar, and backward reference searching
Search window	2015-2025; final audit completed on 4 May 2026
Search terms	AIOps; AI in IT operations; MLOps; ITIL; ITSM; change management; release management; DevOps; public-sector AI; AI governance
Inclusion criteria	Sources addressing AI-enabled IT operations, change/release governance, MLOps, DevOps deployment, public-sector AI adoption, or responsible AI controls

Criteria	Description
Exclusion criteria	Non-English sources, non-technical opinion pieces, duplicated studies, and publications unrelated to operational governance

Table 2: Boolean search strings, search fields, and audit details

Database	Fields searched	Boolean search string
IEEE Xplore	Title, abstract, keywords, metadata; full text when available	("AIOps" OR "artificial intelligence for IT operations" OR "machine learning for IT operations") AND ("IT service management" OR ITSM OR ITIL OR "change management" OR "release management" OR DevOps OR "continuous delivery")
ACM Digital Library	Title, abstract, keywords, metadata	("AIOps" OR "failure management" OR "anomaly detection" OR "root cause analysis") AND (logs OR metrics OR incidents OR "IT operations")
ScienceDirect	Title, abstract, keywords	("artificial intelligence" OR AI OR "machine learning") AND ("public governance" OR "public sector" OR "public administration" OR "government IT") AND ("service management" OR "change management" OR "release management")
Google Scholar	Title and full-text metadata returned by the search engine	(AIOps OR MLOps OR "AI in IT operations") AND (ITIL OR ITSM OR "change management" OR "release management") AND ("public sector" OR government OR governance)

2.1 Research Design

The initial database search identified 120 records. After removing 20 duplicates, 100 records were screened by title, abstract, and keyword relevance. Forty-five records were excluded at screening because they did not address AI-enabled IT operations, service governance, operational automation, or public-sector relevance. Fifty-five full-text articles and sources were then assessed for eligibility; 16 were removed because they lacked methodological clarity, technical depth, or relevance to change/release governance. The final synthesis included 39 sources, consisting of 25 primary studies and 14 supplementary methodological, standards, and governance sources. Figure 1 summarizes the study-selection process and allows readers to verify the flow from identification to final inclusion.

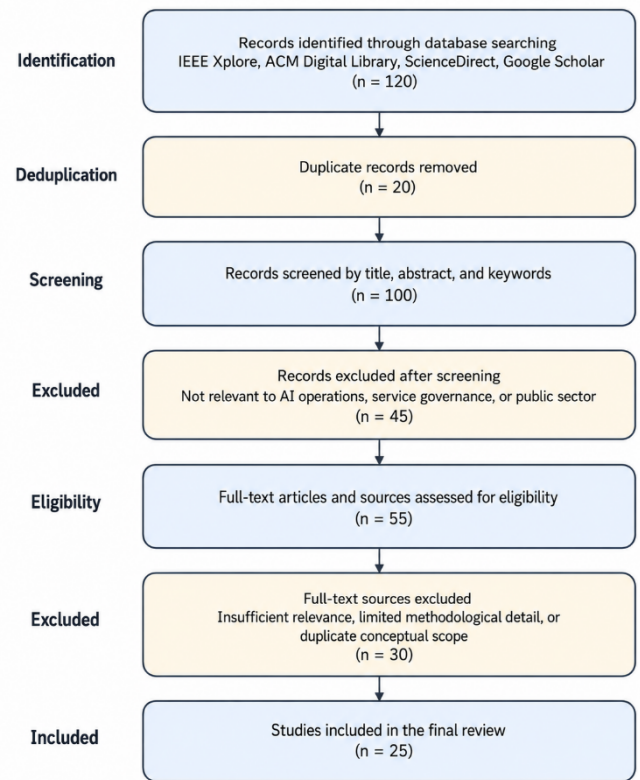


Figure 1: PRISMA-based study selection process.

2.2 Data Extraction, Bias Mitigation, and Analysis

For each selected source, data were extracted on publication year, application domain, AI technique, operational process, governance issue, empirical setting, and relevance to public-sector change and release management. The analysis used thematic coding to identify recurring patterns across the literature, followed by cross-checking themes for coherence and relevance to the research questions. The coding process generated four synthesis themes: AI-driven monitoring and incident management, public-sector governance and compliance, legacy-system and data integration barriers, and AI support for change and release management.

Because this review was conducted by a single independent author, bias mitigation relied on a documented audit trail rather than inter-reviewer reliability statistics. The audit trail recorded search strings, source databases, inclusion/exclusion reasons, quality-assessment notes, and theme-coding decisions. To reduce screening bias, borderline sources were rechecked against the research questions after full-text reading, and each retained source was classified as direct evidence or adjacent evidence. Direct evidence referred to sources that explicitly discussed AIOps, MLOps, ITSM, change management, release governance, or public-sector AI. Adjacent evidence referred to sources on ML production readiness, explainability, fairness, datasets, risk management, and AI management systems. This distinction prevented overstatement of the maturity of

AI-enabled change approval while still incorporating controls needed for responsible implementation.

2.3 Quality Assessment and Coding Controls

Each source was assessed against a compact quality checklist. The checklist examined whether the source clearly stated its problem context, described the AI or governance method, provided transferable evidence for IT operations, and addressed risks such as data quality, explainability, organizational readiness, or auditability. Sources that were strong in AIOps but weak in public-sector context were retained only when their findings were directly transferable to change, release, monitoring, or operational-risk management. Standards and governance sources were used as complementary evidence rather than as primary empirical studies.

Table 3: Quality assessment checklist for retained literature

Assessment area	Question applied during screening	Use in synthesis
Topical relevance	Does the source address AI-enabled operations, ITSM, MLOps, DevOps release control, or public-sector AI governance?	Determined whether the source contributed to one or more research questions.
Method clarity	Does the source explain its review method, empirical setting, framework logic, or technical approach?	Reduced reliance on unclear opinion-based material.
Operational evidence	Does the source discuss logs, incidents, deployment records, change data, service dependencies, or monitoring outcomes?	Supported links between AIOps evidence and ITIL change/release workflows.
Governance relevance	Does the source address transparency, accountability, auditability, risk management, or human oversight?	Informed public-sector requirements for accountable AI adoption.
Transferability	Can findings from enterprise or cloud settings be adapted to regulated government IT environments?	Separated directly applicable findings from contextual limitations.

Table 4.a and 4.b summarizes the quality profile of the 25 primary studies used as the main analytic evidence base. Ratings are reported as H = high, M = moderate, and L = low. The table is intended to make the evidence base

transparent rather than to exclude all studies that scored low on one dimension; for example, a technical AIOps study could score low on governance relevance but still provide strong operational evidence.

Table 4a: Quality profile of technical and operational studies used for thematic synthesis

Ref.	Focus	T	M	O	G	Role in review
[3]	ITSM review	H	H	M	M	ITSM foundation
[4]	AIOps failure management	H	H	H	M	AIOps methods
[5]	AIOps real-world challenges	H	M	H	M	Implementation barriers
[6]	Cloud AIOps	H	H	H	M	Cloud observability
[7]	ML for systems/networking	H	H	H	L	Technical evidence
[8]	MLOps architecture	H	H	M	M	ML lifecycle control
[13]	Agentic AIOps framework	H	M	H	M	ITIL-aligned automation
[14]	ML technical debt	H	H	M	M	ML governance risk
[15]	ML production readiness	H	H	M	M	Model validation control
[20]	Software engineering for ML	H	H	M	M	ML engineering practice
[21]	ML systems challenges	H	H	M	M	Deployment challenges
[22]	ML deployment survey	H	H	M	M	Operational ML risks
[23]	Anomaly and RCA survey	H	H	H	L	Incident analysis
[24]	Log anomaly detection	H	H	H	L	Log-based monitoring
[25]	Time-series anomaly detection	H	H	H	L	Telemetry monitoring
[26]	LLMs for anomaly detection	M	H	H	L	Emerging AI monitoring
[27]	CI/CD systematic review	H	H	H	M	Release pipeline practices
[28]	DevOps adoption case study	H	H	M	M	Organizational barriers

T = topical relevance; M = methodological clarity; O = operational evidence; G = governance relevance. H = high, M = moderate, L = low.

Table 4b: Quality profile of public-sector and governance studies used for thematic synthesis

Ref.	Focus	T	M	O	G	Role in review
[9]	AI in public governance	H	H	L	H	Public-sector governance
[10]	AI and public sector	H	H	L	H	Public-sector challenges
[11]	AI change in public administration	H	M	L	H	Organizational readiness
[12]	AI-enabled organizational change	M	H	L	M	Change adoption lens
[29]	AI adoption in public administration	H	H	L	H	Government adoption evidence
[30]	AI capability in government	H	H	L	H	Public-sector AI capability
[31]	AI public management framework	H	M	L	H	Governance framework

T = topical relevance; M = methodological clarity; O = operational evidence; G = governance relevance. H = high, M = moderate, L = low.

Overall, the quality assessment shows strong topical relevance across the selected studies. Operational evidence is strongest in AIOps, anomaly detection, DevOps, and observability studies, while governance relevance is strongest in public-sector AI and responsible-AI literature. This distribution supports the review's central finding that technical AIOps research is comparatively mature, whereas governance-oriented applications in public-sector change and release management remain less developed

3. Overview of Selected Publications

The reviewed literature shows a strong concentration of work on AIOps for operational monitoring, anomaly detection, and incident response. A smaller body of work addresses public-sector AI governance, organizational readiness, and explainability. Very few studies directly examine how AI can be embedded in ITIL change and release management. Table 4 summarizes representative studies used to build the thematic synthesis.

Additional ML engineering studies make clear that AI-enabled software lifecycle management requires workflows that differ from traditional software development: data collection, model training, evaluation, deployment, monitoring, and feedback loops must be coordinated [20-22]. AIOps literature also shows that log analysis, anomaly detection, root-cause analysis, and multivariate time-series monitoring are mature research areas, but they are typically evaluated for incident response rather than formal change approval [23-26]. DevOps and continuous deployment research provide a process foundation for release governance, emphasizing

automated testing, pipeline visibility, security, and cross-functional collaboration [27], [28]. Public-administration studies add constraints: AI adoption depends on public value tensions, absorptive capacity, organizational capabilities, and risk-aware governance [29-33].

Table 5: Representative primary and complementary studies

Ref(s)	Focus area	Key contribution to this review
[4], [5]	AIOps failure management	Anomaly detection, event correlation, and real-world implementation challenges.
[6], [23]	Cloud and microservice AIOps	Incident detection, failure prediction, root-cause analysis, and cloud observability.
[8], [20]-[22]	MLOps and ML engineering	Lifecycle control, deployment readiness, monitoring, collaboration, and production maintenance.
[24]-[26]	Logs and anomaly detection	Deep log analysis, multivariate time-series detection, and emerging LLM-based methods.
[27], [28]	DevOps and release pipelines	CI/CD practices, automated testing, pipeline visibility, and cultural barriers.
[9]-[12], [29]-[33]	Public-sector AI governance	Transparency, accountability, institutional readiness, and organizational change.
[34]-[39]	Responsible AI controls	Model cards, dataset documentation, fairness, risk management, and AI management-system requirements.

4. Findings and Discussion

The thematic analysis shows that AI has strong operational value in monitoring and incident response, but weaker integration with formal governance processes. Figure 2 summarizes the main classification of AI in IT operations used in this review. The figure separates technical AIOps capabilities from governance-oriented ITIL touchpoints so that the reader can see where current research is mature and where change/release management remains underdeveloped.

4.1 AI-Driven Monitoring and Incident Management

The most mature area of AI-enabled IT operations is monitoring and incident management. AIOps studies demonstrate how machine learning can process logs, alerts, traces, and performance metrics to identify anomalies, correlate related events, and recommend likely root causes [4-6], [23-25]. These capabilities reduce manual triage and support a shift from reactive operations toward predictive service management. For

change and release management, the same telemetry can provide baseline behavior before deployment, early detection after deployment, and evidence for rollback or remediation decisions.

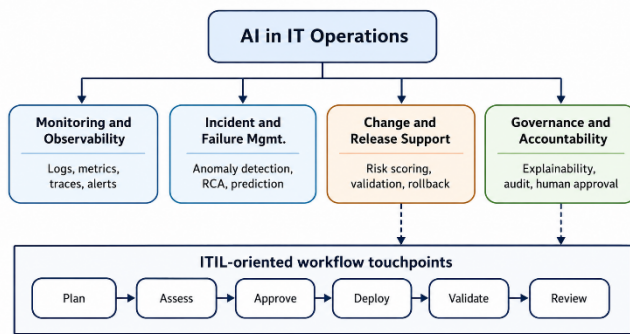


Figure 2: Classification of AI capabilities relevant to IT operations and ITIL workflows

4.2 Change and Release Management as a Governance Bottleneck

Change and release workflows remain less automated than monitoring workflows. Traditional ITIL processes rely on predefined approval paths and human assessment of change risk. DevOps and CI/CD research show that automated testing, deployment pipelines, and continuous feedback can improve release frequency and reliability [27], [28]. However, public-sector environments often require additional approval controls because releases may affect regulated services, public records, security controls, or statutory reporting. The review therefore indicates that AI should be used as decision support for risk scoring, schedule optimization, dependency analysis, rollback readiness, and post-release validation rather than as an unchecked autonomous approver.

4.3 Public-Sector Governance, Compliance, and Accountability

Public-sector AI adoption is shaped by accountability, fairness, transparency, privacy, legal compliance, procurement rules, and public trust [9-12], [29-33]. These constraints are directly relevant to AI-enabled change management because the output of a risk-scoring model may influence whether a system change is approved, delayed, or escalated. Governance therefore requires evidence that a model was trained on appropriate data, evaluated under relevant conditions, monitored after deployment, and reviewed by accountable personnel.

Responsible-AI literature provides mechanisms for making operational AI more auditable. Model cards document intended use, evaluation conditions, limitations, and ethical considerations [34]. Dataset datasheets document data provenance, composition, collection processes, and recommended use [35]. Bias and

fairness studies show that model behavior can be affected by data quality and hidden assumptions, while fairness toolkits provide techniques for detection and mitigation [36], [37]. Governance frameworks such as the NIST AI Risk Management Framework and ISO/IEC 42001 translate these principles into lifecycle controls for risk management, monitoring, documentation, and continual improvement [38], [39].

4.4 Legacy Systems, Data Quality, and Integration Barriers

A recurring barrier across AIOps and MLOps literature is the quality and integration of operational data. AIOps depend on reliable logs, metrics, alerts, incidents, configuration data, deployment records, and service dependencies [5], [6]. Public-sector agencies often operate a mixture of legacy applications, custom platforms, cloud services, and vendor systems, which can make data normalization and real-time correlation difficult. MLOps literature reinforces that AI models require monitoring, validation, and retraining after deployment [8], [20-22]. Without these controls, models used for change-risk prediction may become inaccurate as infrastructure, data patterns, and service dependencies change.

4.5 AI-Enabled ITIL Change and Release Framework

Based on the synthesis, this review proposes a framework that links operational data, AIOps analytics, MLOps governance, ITIL change/release processes, and public-sector oversight. The framework is not intended to remove human control. Instead, it shows where AI can strengthen evidence-based decision-making while keeping accountability, explainability, and compliance within the governance process. Figure 3 presents the proposed framework.

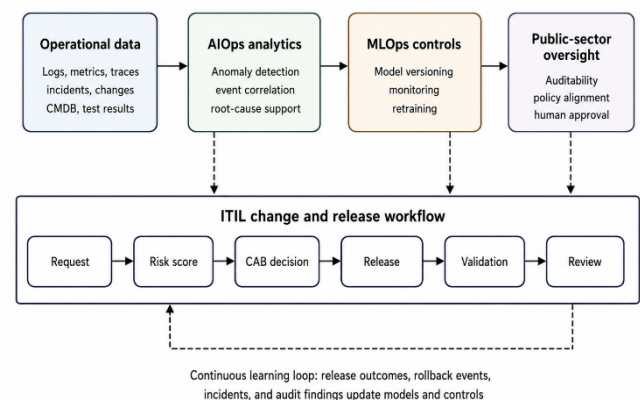
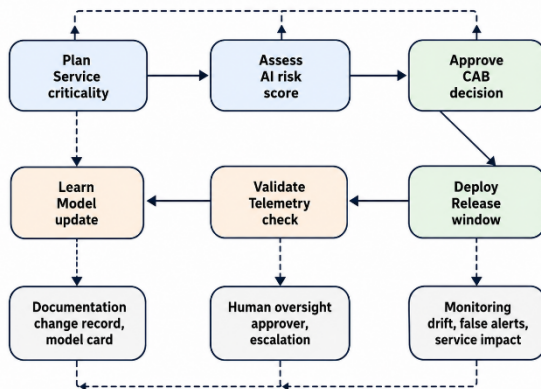


Figure 3: Proposed AI-enabled ITIL change and release management framework

The framework can be operationalized through lifecycle controls that connect planning, assessment, approval, deployment, validation, and review. Figure 4

shows how these controls distribute across the AI-enabled release process. It also clarifies that governance controls do not occur only at approval time; they appear before, during, and after deployment.



Governance loop: deployment evidence informs future risk assessment and CAB review

Figure 4: Governance controls across the AI-enabled release lifecycle

4.6 Design Requirements for AI-Assisted CAB Decisions

The synthesis indicates that an AI-assisted CAB should be designed as an accountable decision-support capability rather than a replacement for review boards. The system should combine operational telemetry with change records, but it must also expose why a recommendation was made and whether the supporting evidence is complete. Table 6 translates the literature into design requirements for public-sector use.

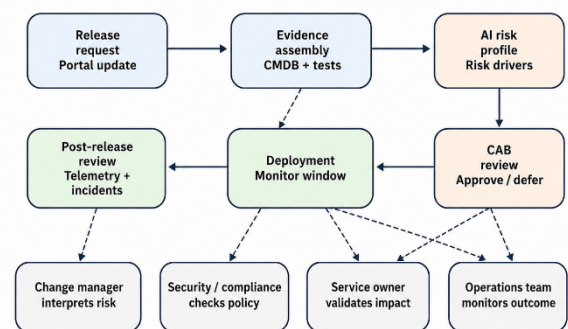
Table 6: Design requirements for AI-assisted change advisory decisions

Requirement	Rationale	Implementation controls
Explainable risk score	CAB members need to understand why change is low, medium, or high risk.	Show risk drivers, affected services, historical incidents, dependency confidence, and uncertainty.
Data provenance	Operational data may be fragmented across legacy systems and vendor platforms.	Track source systems, time windows, missing values, normalization logic, and dataset limits.
Human approval	Public-sector accountability cannot be delegated fully to an algorithm.	Require named approver, escalation path, and policy-based exception handling.
Release evidence	Approval should be connected to testing and deployment readiness.	Link test results, security checks, rollback plan, change window, and service-owner sign-off.
Post-release learning	The model should improve from release outcomes	Feed incidents, rollback events, performance shifts,

Requirement	Rationale	Implementation controls
	without hiding errors.	and user-impact data into review cycles.
Model governance	AIOps models can drift or become misaligned with policy requirements.	Use monitoring, retraining triggers, model cards, dataset documentation, and periodic audit review.

4.7 Illustrative Public-Sector Workflow Simulation

To make the proposed framework more concrete, consider an illustrative public-sector portal release. A change manager submits a release request for an online regulatory portal. The AI-assisted workflow first assembles evidence from the change record, impacted configuration items, prior incidents, test results, and service criticality. An AIOps model then generates a risk profile that identifies likely service dependencies, unusual historical outage patterns, and confidence limits in the available data. The CAB does not automatically accept the recommendation; instead, it uses the risk profile to ask targeted questions about rollback readiness, affected services, and compliance risk. During deployment, telemetry is monitored against historical baselines. After deployment, incidents, rollbacks, or abnormal performance shifts feed into the post-implementation review and future model retraining. Figure 5 summarizes this scenario as a decision-support workflow.



The workflow demonstrates decision support rather than autonomous approval in regulated public-sector environments.

Figure 5: Illustrative public-sector workflow simulation for a portal release

5. Research gaps and Future Directions

The literature provides strong foundations for AIOps, MLOps, DevOps, and responsible AI, but several gaps remain when these areas are applied to public-sector change and release management. Table 7 summarizes the main gaps and future research directions.

Table 7: Research gaps and future directions

Gap	Problem	Future direction
Limited empirical evidence	Few studies evaluate AI in real public-sector change/release workflows.	Conduct case studies using operational change records, incidents, release outcomes, and audit findings.
Weak explainability for CAB decisions	Risk scores alone are insufficient for accountable approval.	Develop explanation methods that show risk drivers, data confidence, affected services, and rollback readiness.
AI-ready change records	Change tickets often lack structured features needed for prediction.	Standardize change metadata: affected CI, dependency, test result, outage history, approval path, and service criticality.
Governance-aware MLOps	AIOps models are rarely connected to public-sector policy controls.	Map model monitoring, retraining, validation, access control, and documentation to ITIL and AI governance requirements.
Human-AI collaboration	Automation can reduce effort but may reduce accountability if poorly designed.	Evaluate decision-support interfaces for CABs, release managers, service owners, and compliance reviewers.
Transferability across agencies	Public agencies differ in infrastructure, law, data maturity, and risk tolerance.	Build adaptable frameworks with maturity levels rather than one-size-fits-all automation models.

Future research should move from general AIOps capability descriptions toward process-specific evaluation. Public-sector studies should examine where AI recommendations can safely enter the change workflow, which decisions remain strictly human-controlled, and how agencies can measure improvement in release risk, incident reduction, audit readiness, and service continuity. Research should also examine how MLOps controls can be embedded into ITSM so that operational AI models remain reliable after infrastructure, data, and policy conditions change.

A practical research direction is the development of an AI-assisted CAB decision-support model. Such a model would combine change history, dependency maps, incident records, test results, service criticality, and compliance requirements to produce an explainable

change-risk profile. The profile should support, not replace, human decision-making by presenting risk factors, uncertainty, relevant precedent, and recommended mitigation actions. For public agencies, this design is preferable to black-box automation because it preserves accountability while improving evidence quality.

Another research direction concerns evaluation of metrics. Existing AIOps studies often evaluate detection accuracy or alert reduction, while public-sector change governance also requires measures such as auditability, approval transparency, incident avoidance, service-continuity protection, and stakeholder confidence. Future evaluations should therefore combine technical measures with organizational and governance outcomes.

6. Practical Implications for Public-Sector IT

For public-sector agencies, the findings suggest that AI-enabled change and release management should be introduced through a staged maturity path. Agencies should first improve the quality of change records, service maps, incident histories, and release metrics before relying on predictive models. Without these foundations, AI outputs may appear precise while reflecting incomplete data. The practical starting point is therefore not automation of approvals; it is improving operational observability and creating reliable evidence for human review.

A second implication is that AI should be embedded into existing ITIL and governance roles rather than deployed as a separate analytics experiment. Change managers, release managers, service owners, security teams, data stewards, and compliance reviewers should have defined responsibilities in the AI-enabled workflow. This role clarity is important because public-sector releases often involve statutory obligations, vendor dependencies, security controls, and citizen-facing services. A risk score is useful only when the organization knows who must interpret it, who can override it, and how the decision is documented.

A third implication concerns procurement and vendor management. Many agencies adopt monitoring, cloud, ITSM, and AIOps tools through vendor platforms. Procurement language should require data exportability, audit logs, model documentation, access controls, and explainability features. These requirements help prevent dependency on opaque vendor models and support later integration with agency-specific ITIL workflows, risk registers, and compliance evidence.

Table 8 summarizes a phased implementation roadmap for AI-enabled change and release management in public-sector IT. The roadmap highlights how agencies

can move from foundational data readiness and observability toward predictive decision support, governance integration, and continuous improvement. This staged approach ensures that AI adoption remains aligned with ITIL workflows, human oversight, compliance expectations, and operational reliability.

Table 8: Implementation roadmap for AI-enabled change and release management

Phase	Key actions	Expected outcome
Phase 1: Data readiness	Standardize change tickets, incident taxonomies, configuration items, and service criticality.	Cleaner operational evidence and more reliable baseline metrics.
Phase 2: Observability	Connect logs, metrics, traces, alerts, test outcomes, and deployment records.	Earlier detection of abnormal release behavior and stronger post-release validation.
Phase 3: Predictive support	Pilot change-risk scoring using historical releases and incident outcomes.	Evidence-based CAB discussion without removing human approval.
Phase 4: Governance integration	Map AI outputs to ITIL controls, approval policies, model cards, dataset records, and audit trails.	Traceable decisions and stronger compliance readiness.
Phase 5: Continuous improvement	Monitor model drift, review false positives/negatives, update training data, and evaluate agency outcomes.	Improved reliability, reduced release risk, and sustainable AI operations.

The roadmap also defines measurable outcomes for pilot work. Agencies should evaluate the reduction in post-release incidents, mean time to detect abnormal behavior, false-positive and false-negative risk alerts, CAB cycle time, rollback preparedness, and audit evidence completeness. These metrics are more useful than broad claims about automation because they connect AI adoption to public-sector service reliability and governance outcomes.

7. Limitations

This review has three limitations. First, the available literature contains more studies on monitoring, anomaly detection, and incident response than on AI-enabled change approval or release governance. As a result, some findings are transferred from adjacent areas such as MLOps, DevOps, cloud observability, and public-sector AI governance. Second, the review emphasizes peer-reviewed and high-quality governance sources, but

implementation practices in agencies may also be documented in internal playbooks, procurement records, and vendor reports that are not publicly available. Third, the proposed framework is conceptual and should be validated through agency-level case studies, interviews, and pilot deployments.

These limitations do not weaken the main finding; rather, they clarify where empirical work is needed. The literature is sufficiently mature to identify the technical and governance components required for AI-assisted change and release management, but it is not yet mature enough to claim that a single reference architecture will fit all public-sector agencies. Future validation should therefore compare different agency sizes, risk profiles, service portfolios, and data maturities.

8. Conclusion

This systematic literature review examined the transition from ITIL-based service governance toward AI-enabled operations in public-sector IT. The review shows that AIOps is well developed for monitoring, anomaly detection, incident response, and failure prediction. MLOps provides complementary controls for model deployment, validation, monitoring, and retraining. Public-sector AI literature emphasizes transparency, accountability, governance, and organizational readiness.

The central finding is that AI has not yet been sufficiently integrated into formal change and release management. Intelligent monitoring provides useful technical foundations, but public-sector agencies need validated frameworks that connect operational analytics with ITIL workflows, human approval, compliance requirements, and explainable recommendations. The proposed framework positions AI as a decision-support layer that can improve risk assessment, release planning, and post-release validation while preserving governance accountability.

For practice, the review suggests that public-sector IT leaders should begin with high-value, low-risk AI use cases such as anomaly detection, post-release monitoring, and evidence-based change-risk assessment. For research, the next step is empirical validation through public-sector case studies, pilot implementations, and standard evaluation metrics that determine how AI can improve service continuity, operational efficiency, and trustworthy governance in public-sector IT environments.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgement

The author thanks the reviewers for their constructive feedback, which helped improve the methodological transparency, figure explanations, and practical framing of this review.

References

- [1] AXELOS, ITIL Foundation: ITIL 4 Edition. London, U.K.: The Stationery Office, 2019.
- [2] J. Iden and T. R. Eikebrokk, "Implementing IT Service Management: A systematic literature review," *International Journal of Information Management*, vol. 33, no. 3, pp. 512-523, 2013, doi:10.1016/j.ijinfomgt.2013.01.004.
- [3] J. Serrano, J. Faustino, D. Adriano, R. Pereira, and M. M. da Silva, "An IT Service Management Literature Review: Challenges, Benefits, Opportunities and Implementation Practices," *Information*, vol. 12, no. 3, article 111, 2021, doi:10.3390/info12030111.
- [4] P. Notaro, J. Cardoso, and M. Gerndt, "A Survey of AIOps Methods for Failure Management," *ACM Transactions on Intelligent Systems and Technology*, vol. 12, no. 6, article 81, pp. 1-45, 2021, doi:10.1145/3483424.
- [5] Y. Dang, Q. Lin, and P. Huang, "AIOps: Real-World Challenges and Research Innovations," in 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2019, pp. 4-5, doi:10.1109/ICSE-Companion.2019.00023.
- [6] Q. Cheng et al., "AI for IT Operations (AIOps) on Cloud Platforms: Reviews, Opportunities and Challenges," *arXiv:2304.04661*, 2023, doi:10.48550/arXiv.2304.04661.
- [7] M. E. Kanakis, R. Khalili, and L. Wang, "Machine Learning for Computer Systems and Networking: A Survey," *ACM Computing Surveys*, vol. 55, no. 4, article 71, pp. 1-36, 2022, doi:10.1145/3523057.
- [8] D. Kreuzberger, N. Kuehl, and S. Hirschl, "Machine Learning Operations (MLOps): Overview, Definition, and Architecture," *IEEE Access*, vol. 11, pp. 31866-31879, 2023, doi:10.1109/ACCESS.2023.3262138.
- [9] A. Zuiderwijk, Y.-C. Chen, and F. Salem, "Implications of the use of artificial intelligence in public governance: A systematic literature review and a research agenda," *Government Information Quarterly*, vol. 38, no. 3, article 101577, 2021, doi:10.1016/j.giq.2021.101577.
- [10] B. W. Wirtz, J. C. Weyerer, and C. Geyer, "Artificial Intelligence and the Public Sector: Applications and Challenges," *International Journal of Public Administration*, vol. 42, no. 7, pp. 596-615, 2019, doi:10.1080/01900692.2018.1498103.
- [11] I. Mainardi, "Change management: artificial intelligence (AI) at the service of public administrations," *AI & Society*, vol. 40, no. 5, pp. 3953-3981, 2025, doi:10.1007/s00146-024-02136-2.
- [12] J. Schwaake, C. Gerlich, H. L. Nguyen, D. K. Kanbach, and J. Gast, "Artificial intelligence (AI) for good? Enabling organizational change towards sustainability," *Review of Managerial Science*, vol. 19, pp. 3013-3038, 2025, doi:10.1007/s11846-025-00840-x.
- [13] R. D. Zota, C. Barbulescu, and R. Constantinescu, "A Practical Approach to Defining a Framework for Developing an Agentic AIOps System," *Electronics*, vol. 14, no. 9, article 1775, 2025, doi:10.3390/electronics14091775.
- [14] D. Sculley et al., "Hidden Technical Debt in Machine Learning Systems," in *Advances in Neural Information Processing Systems* 28, 2015, pp. 2503-2511.
- [15] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction," in 2017 IEEE International Conference on Big Data, 2017, pp. 1123-1132, doi:10.1109/BigData.2017.8258038.
- [16] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138-52160, 2018, doi:10.1109/ACCESS.2018.2870052.
- [17] M. J. Page et al., "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," *BMJ*, vol. 372, article n71, 2021, doi:10.1136/bmj.n71.
- [18] C. Okoli, "A Guide to Conducting a Standalone Systematic Literature Review," *Communications of the Association for Information Systems*, vol. 37, article 43, 2015, doi:10.17705/1CAIS.03743.
- [19] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Research in Psychology*, vol. 3, no. 2, pp. 77-101, 2006, doi:10.1191/1478088706qp0630a.
- [20] S. Amershi et al., "Software Engineering for Machine Learning: A Case Study," in 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), 2019, pp. 291-300, doi:10.1109/ICSE-SEIP.2019.00042.
- [21] L. E. Lwakatare, A. Raj, J. Bosch, H. H. Olsson, and I. Crnkovic, "A Taxonomy of Software Engineering Challenges for Machine Learning Systems: An Empirical Investigation," in *Agile Processes in Software Engineering and Extreme Programming*, Cham, Switzerland: Springer, 2019, pp. 227-243, doi:10.1007/978-3-030-19034-7_14.
- [22] A. Paleyes, R.-G. Urma, and N. D. Lawrence, "Challenges in Deploying Machine Learning: A Survey of Case Studies," *ACM Computing Surveys*, vol. 55, no. 6, article 114, pp. 1-29, 2022, doi:10.1145/3533378.
- [23] J. Soldani and A. Brogi, "Anomaly Detection and Failure Root Cause Analysis in (Micro)Service-Based Cloud Applications: A Survey," *ACM Computing Surveys*, vol. 55, no. 3, article 59, pp. 1-39, 2022, doi:10.1145/3501297.
- [24] M. Du, F. Li, G. Zheng, and V. Srikumar, "DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1285-1298, doi:10.1145/3133956.3134015.
- [25] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019, pp. 2828-2837, doi:10.1145/3292500.3330672.
- [26] J. Su et al., "Large Language Models for Forecasting and Anomaly Detection: A Systematic Literature Review," *arXiv:2402.10350*, 2024, doi:10.48550/arXiv.2402.10350.
- [27] M. Shahin, M. A. Babar, and L. Zhu, "Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices," *IEEE Access*, vol. 5, pp. 3909-3943, 2017, doi:10.1109/ACCESS.2017.2685629.
- [28] L. Riungu-Kalliosaari, S. Mäkinen, L. E. Lwakatare, J. Tiihonen, and T. Männistö, "DevOps Adoption Benefits and Challenges in Practice: A Case Study," in *Product-Focused Software Process Improvement*, Cham, Switzerland: Springer, 2016, pp. 590-597, doi:10.1007/978-3-319-49094-6_44.
- [29] R. Madan and M. Ashok, "AI adoption and diffusion in public administration: A systematic literature review and future research agenda," *Government Information Quarterly*, vol. 40, no. 1, article 101774, 2023, doi:10.1016/j.giq.2022.101774.
- [30] P. Mikalef et al., "Enabling AI capabilities in government agencies: A study of determinants for European municipalities," *Government Information Quarterly*, vol. 39, no. 4, article 101596, 2022, doi:10.1016/j.giq.2021.101596.

- [31] B. W. Wirtz and W. M. Müller, "An integrated artificial intelligence framework for public management," *Public Management Review*, vol. 21, no. 7, pp. 1076-1100, 2019, doi:10.1080/14719037.2018.1549268.
- [32] J. Berryhill, K. K. Heang, R. Clogher, and K. McBride, *Hello, World: Artificial Intelligence and Its Use in the Public Sector*. Paris, France: OECD Publishing, 2019, doi:10.1787/726fd39d-en.
- [33] OECD, *Recommendation of the Council on Artificial Intelligence*, OECD/LEGAL/0449, Paris, France: OECD, 2019.
- [34] M. Mitchell et al., "Model Cards for Model Reporting," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019, pp. 220-229, doi:10.1145/3287560.3287596.
- [35] T. Gebru et al., "Datasheets for Datasets," *Communications of the ACM*, vol. 64, no. 12, pp. 86-92, 2021, doi:10.1145/3458723.
- [36] 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, article 115, pp. 1-35, 2021, doi:10.1145/3457607.
- [37] 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, pp. 1-15, 2019, doi:10.1147/JRD.2019.2942287.
- [38] E. Tabassi, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, NIST AI 100-1. Gaithersburg, MD, USA: National Institute of Standards and Technology, 2023, doi:10.6028/NIST.AI.100-1.
- [39] ISO/IEC, *ISO/IEC 42001:2023, Information Technology - Artificial Intelligence - Management System*. Geneva, Switzerland: International Organization for Standardization, 2023.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



CATHERINE GANDURI is a Systems Analyst at the Texas Commission on Environmental Quality (TCEQ), USA. She holds a Master's degree in Business Analytics from The University of Texas at Dallas. Her research interests include Artificial

Intelligence in IT Operations (AIOps), IT Service Management (ITIL), change and release management, and digital transformation in public sector systems. She has experience working with data analytics, enterprise systems, and process optimization to improve operational efficiency.

Graph Neural Networks for Fault Diagnostics in Cyber-Physical Systems: A Survey of Taxonomy, Deployment Architectures and Failure Modes

Vaibhavi Tiwari^{*,1}, Ola Suaifan¹, Ramy Othman^{*,1}, Anand Gupta²

¹School of Computing, Montclair State University, Montclair, NJ 07043, USA

²AWS Corporation, USA

Email(s): tiwariv1@montclair.edu (V. Tiwari), suaifano1@montclair.edu (O. Suaifan), othmanr@montclair.edu (R. Othman), anagupta@amazon.com (A. Gupta)

*Corresponding authors: Vaibhavi Tiwari, Ramy Othman

Email: tiwariv1@montclair.edu, othmanr@montclair.edu

ABSTRACT: Graph Neural Networks (GNNs) have emerged as a promising approach for fault diagnosis in complex cyber-physical systems because they can model intercomponent relationships, fault propagation, and system-level anomalies across domains such as industrial automation, smart grids, transportation, and healthcare. This survey presents a multidimensional review of GNN-based fault diagnostics, organizing existing methods according to graph representation, learning paradigm, diagnostic objective, and deployment context. It examines commonly used benchmark datasets, evaluation protocols, and cloud, edge, hybrid, and federated deployment architectures, with particular attention to reproducibility and practical implementation. In addition to methodological limitations, the survey identifies operational failure modes, including cascading misdiagnosis, topology drift, noise amplification, open-set misclassification, adversarial vulnerability, and concept drift, and examines their implications for safety-critical systems. Emerging research directions, including physics-informed learning, multimodal fusion, dynamic graph modeling, and privacy-preserving federated GNNs, are discussed alongside their ethical and safety implications. Unlike reviews centered primarily on diagnostic tasks or application domains, this survey integrates methodological, architectural, and operational perspectives within a unified framework. The analysis indicates that although GNNs offer capabilities for dependency-aware fault diagnosis, their practical deployment remains constrained by inconsistent benchmarking, sensitivity to graph construction, computational requirements, limited interpretability, and insufficient validation under evolving operational conditions. Finally, practitioner-oriented design guidelines are presented to support the development of reliable, robust, and deployable GNN-based diagnostic systems and to connect algorithmic advances with the operational reliability requirements of next-generation fault-tolerant infrastructures.

KEYWORDS: Graph Neural Networks, Fault Diagnostics, Cyber-Physical Systems, Fault Tolerance, Reliability Engineering, Predictive Maintenance

1. Introduction

Fault diagnostics plays a central role in maintaining the reliability, safety, and operational continuity of modern cyber-physical systems, including industrial automation, smart grids, transportation networks, and healthcare monitoring platforms. As these systems become increasingly complex and interconnected, failures may propagate across components rather than remain isolated, resulting in cascading effects and system-wide disruptions. Recent industry analyses indicate the scale of this problem: average downtime costs exceed \$14,056 per minute, 55–61% of manufacturers experience unplanned downtime annually, and 66–80% of failures are attributed to human error. Globally, downtime losses are estimated to exceed \$400 billion per year [1, 2], highlighting the economic and operational risks associated with inadequate fault tolerance. These trends indicate that improving system resilience is both a technical and operational priority.

Traditional fault diagnostic approaches have relied on physics-based models, signal processing techniques, and statistical methods. Although effective in well-understood environments, these approaches often struggle to scale to high-dimensional, sensor-rich systems characterized by nonlinear dependencies and dynamic interactions [3]. The development of machine learning, particularly deep learning methods such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), has improved diagnostic performance through automated feature extraction from time-series and vibration signals. However, these methods generally assume grid-like data structures and may not adequately capture relational dependencies among system components. As modern infrastructures evolve into highly interconnected systems, modeling techniques are required that can explicitly represent component interactions, failure propagation pathways, and system-wide dependencies.

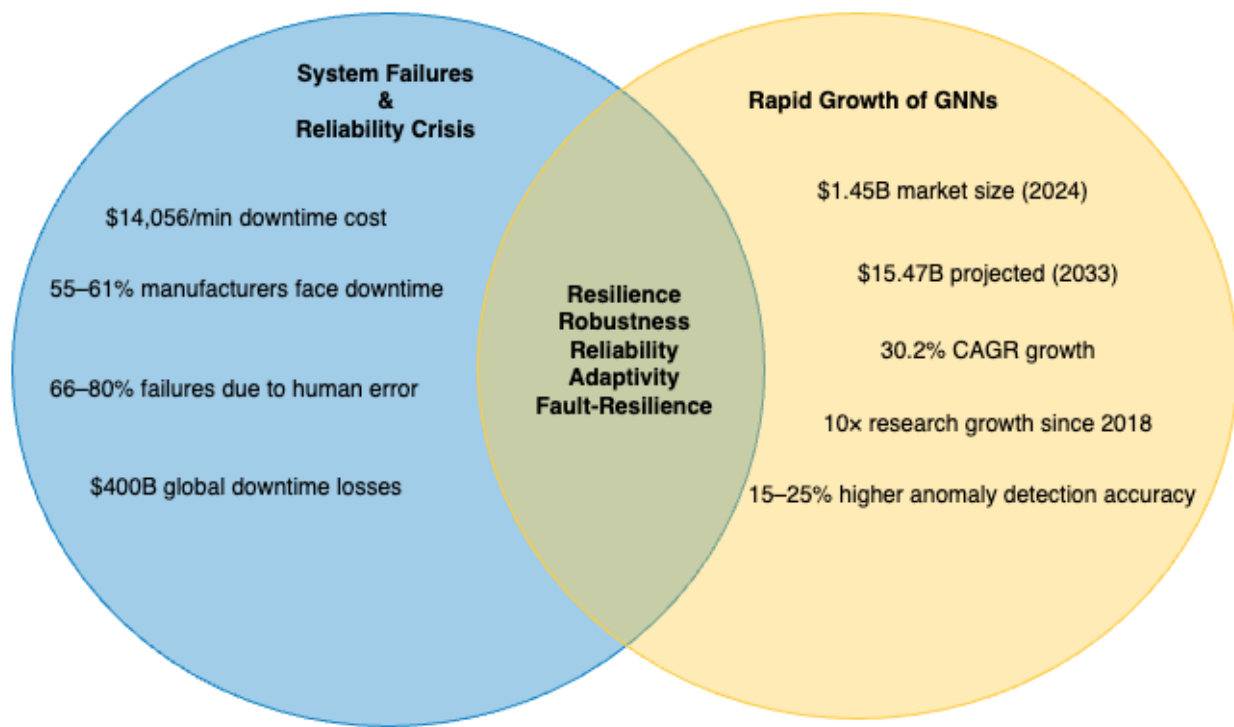


Figure 1: Intersection of system reliability challenges and the growth of Graph Neural Network (GNN) research, motivating resilience-oriented fault diagnostics.

Graph-based modeling provides a natural representation for complex systems, in which nodes correspond to sensors or components and edges represent physical connections, functional dependencies, or statistical correlations. Building on this representation, Graph Neural Networks (GNNs) have demonstrated the ability to learn from structured relational data through message-passing and neighborhood-aggregation mechanisms. By explicitly modeling intercomponent relationships, GNNs can support fault localization, improve robustness to noisy or incomplete sensor data, and represent cascading failure patterns. The growth of GNN adoption further reflects their relevance: the global GNN market was valued at \$1.45 billion in 2024 and is projected to reach \$15.47 billion by 2033, corresponding to a 30.2% compound annual growth rate [4, 5]. Concurrently, GNN research output has increased more than tenfold since 2018 [6, 7], and several studies report improvements of 15–25% in anomaly detection accuracy for networked systems compared with traditional machine learning methods [8, 9, 10]. These findings position GNNs as a promising basis for dependency-aware and fault-tolerant diagnostic systems.

Figure 1 illustrates the relationship between reliability challenges in complex systems and the continued development of GNN-based methods. Rising downtime costs, recurring failures, and systemic vulnerabilities coincide with the scalability, relational modeling capability, and predictive potential of GNNs, motivating diagnostic approaches that integrate predictive fault detection, cascading failure modeling, and dependency-aware system design. In recent years, GNN-based fault diagnostics have been investigated across several domains, including spatiotemporal GNNs for rotating machinery monitoring, graph attention mechanisms for power-grid fault localization, dy-

namic graph learning for IoT anomaly detection, and causal intervention-based GNNs for complex industrial processes [7, 8, 11, 12]. Despite these developments, several challenges remain, including reliable graph construction from raw sensor data, limited availability of labeled fault data, real-time deployment constraints on edge devices, and the need for interpretable and causally grounded diagnostic decisions.

This survey presents a structured, system-oriented synthesis of GNN-based fault diagnostics. Specifically, (i) categorizes existing approaches according to graph representation, learning paradigm, diagnostic objective, and deployment context; (ii) analyzes benchmark datasets and evaluation practices, with emphasis on reproducibility challenges; (iii) examines cloud, edge, hybrid, and federated deployment architectures to assess practical feasibility; (iv) characterizes operational failure modes and distinguishes them from methodological limitations in safety-critical settings; and (v) presents practitioner-oriented design guidelines and ethical considerations to support the development of reliable, resilient, and deployable diagnostic systems.

2. Methodology

This survey used a structured retrospective literature-review methodology to synthesize research on Graph Neural Network (GNN)-based fault diagnostics. The review process was documented retrospectively because the initial literature collection was developed iteratively during manuscript preparation rather than through a prospectively registered review protocol. Accordingly, the methodology was informed by the transparency principles of the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) 2020 framework [13], but the survey

is not presented as a fully registered systematic review. The retrospective procedure focused on documenting the information sources, search concepts, eligibility criteria, treatment of duplicate and preprint records, study categorization, and evidence-synthesis process used to construct the final literature base.

2.1. Review Scope and Research Questions

The review examined GNN-based methods for fault detection, classification, localization, anomaly detection, prognostics, remaining useful life estimation, and operational deployment in industrial and cyber-physical systems. The principal publication period covered studies from 2018 to 2026, corresponding to the period in which GNN-based diagnostic research became increasingly established [6, 7, 14, 15]. Earlier studies were retained when they introduced foundational fault-diagnostic methods, graph-learning architectures, benchmark datasets, or evaluation practices required to explain the historical development of the field. The review addressed the following questions: which graph representations and GNN architectures are used for fault diagnostics; which learning paradigms and diagnostic objectives are considered; which datasets, metrics, and validation protocols are used; how GNN-based systems are deployed in cloud, edge, hybrid, federated, distributed, and digital-twin-supported environments; and which methodological limitations, operational failure modes, ethical concerns, and emerging research directions remain insufficiently addressed.

2.2. Information Sources and Search Concepts

Relevant studies were identified through multidisciplinary engineering and computing sources, including IEEE Xplore, Scopus, Web of Science, ScienceDirect, Springer-Link, ACM Digital Library, IOPscience, and Google Scholar. The literature collection was supplemented through backward reference searching of relevant reviews and primary studies and forward citation tracking of foundational and recent publications. Search concepts combined graph-learning terminology with fault-diagnostic, prognostic, application-domain, and deployment terms. Representative graph-related terms included graph neural network, graph convolutional network, graph attention network, spatiotemporal graph neural network, hypergraph, dynamic graph, and heterogeneous graph. Diagnostic terms included fault diagnosis, fault detection, fault classification, fault localization, anomaly detection, prognostics, and remaining useful life. Additional searches incorporated terms related to explainability, physics-informed learning, federated learning, multimodal fusion, open-set recognition, uncertainty quantification, adversarial robustness, edge deployment, and digital twins.

2.3. Eligibility Criteria

Studies were considered eligible when they applied, evaluated, or systematically examined a graph-learning method relevant to fault diagnostics or condition monitoring. Eligible methods included graph convolutional networks, graph attention networks, spatiotemporal GNNs,

dynamic graph models, heterogeneous graphs, hypergraph networks, Graph-Transformer architectures, and related graph-based learning approaches [14, 15, 16]. Studies were required to address at least one diagnostic objective, such as fault detection, classification, localization, anomaly detection, prognostics, or remaining useful life estimation, and to model sensors, components, variables, subsystems, or physical relationships through a graph representation. The scope included industrial manufacturing, rotating machinery, power systems, transportation, aerospace, telecommunications, IIoT, cyber-physical infrastructure, energy systems, and biomedical monitoring. Peer-reviewed journal articles and conference papers were prioritized, while review papers were retained for historical synthesis, terminology, and citation tracking. Dataset papers and institutional repositories were included when they defined benchmarks used in the reviewed literature.

2.4. Exclusion Criteria

Studies were excluded when they used graphs only for visualization or conventional network analysis without a graph-learning model; addressed generic graph-classification tasks without relevance to fault diagnostics or condition monitoring; relied solely on conventional machine-learning or deep-learning methods without graph representation; lacked sufficient methodological or experimental detail; or fell outside the engineering, industrial, infrastructure, or biomedical-monitoring scope of the survey. Abstract-only publications, posters, patents, theses, editorials, and nontechnical commentary were not treated as primary methodological evidence. Commercial market reports and industry webpages were used only to support contextual statistics in the Introduction and were excluded from the methodological synthesis.

2.5. Treatment of Duplicate Records and Preprints

The bibliography was reviewed retrospectively to identify duplicate records, repeated DOI entries, and preprint-publication pairs. When both a preprint and a peer-reviewed journal or conference version of the same study were available, the formally published version was retained. Citation metadata were checked for consistency in title, author list, publication year, venue, volume, article number, page range, and DOI. Preprints were retained only when no corresponding peer-reviewed version was identified and when the study addressed a recent or underrepresented topic relevant to the survey. Such records were explicitly identified as preprints in the bibliography and were interpreted more cautiously than peer-reviewed evidence.

2.6. Study Selection and Retrospective Verification

The final literature base was constructed through iterative identification, relevance screening, full-text assessment, citation checking, and bibliography verification. Titles and abstracts were first examined for relevance to GNN-based fault diagnostics, after which potentially eligible studies were assessed in full text. Studies were retained when they contributed methodological, experimental, deployment,

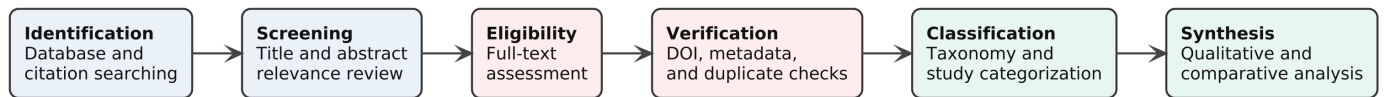


Figure 2: PRISMA-informed retrospective review workflow used to identify, screen, verify, classify, and synthesize literature on GNN-based fault diagnostics. The process includes database and citation searching, title and abstract screening, full-text eligibility assessment, bibliographic verification, taxonomy-based classification, and qualitative comparative synthesis [13].

benchmark, robustness, explainability, or safety-related evidence relevant to the survey taxonomy. During revision, the retained bibliography was re-examined to remove duplicate entries, replace superseded preprints with published versions, verify recent 2025–2026 publications, and distinguish primary studies from reviews, dataset papers, standards, technical reports, and contextual sources. Because complete records of the original database-result counts and sequential exclusion decisions were not preserved, numerical PRISMA counts are not reported. Figure 2 presents the documented retrospective workflow used to identify, screen, verify, classify, and synthesize the retained literature.

2.7. Data Extraction and Categorization

For each retained study, information was extracted on publication year, publication type, application domain, diagnostic objective, graph-construction strategy, node and edge definitions, static or dynamic topology, GNN architecture, learning paradigm, data modality, benchmark dataset, preprocessing method, evaluation metrics, comparison baselines, explainability mechanism, uncertainty-estimation approach, deployment context, computational requirements, robustness evaluation, and reported limitations. The studies were organized according to four principal dimensions: graph structure and representation, learning paradigm, diagnostic objective, and deployment context. Additional synthesis categories covered application domains, benchmark datasets, evaluation protocols, methodological limitations, operational failure modes, ethical and safety concerns, emerging research directions, and practitioner-oriented design considerations.

2.8. Quality and Relevance Assessment

The methodological relevance of the retained studies was evaluated using five criteria: clarity of graph construction and system representation; adequacy of the dataset and experimental design; transparency of preprocessing, training, and validation procedures; appropriateness of evaluation metrics and comparison baselines; and completeness of robustness, deployment, or limitations reporting. Each criterion was assessed qualitatively as fully reported, partially reported, or not reported. Studies were not excluded solely because one reporting category was incomplete when they contributed foundational, historical, benchmark, or emerging methodological information. However, claims derived from studies with incomplete validation or limited experimental transparency were interpreted cautiously.

2.9. Evidence Synthesis

A structured qualitative and taxonomy-based synthesis was conducted because the reviewed studies differed substantially in application domain, graph construction,

datasets, operating conditions, fault definitions, model configurations, train–test procedures, and evaluation metrics. These differences prevented a reliable statistical meta-analysis across the full evidence base. The synthesis therefore compared recurring graph representations, learning paradigms, diagnostic objectives, benchmark practices, deployment architectures, robustness mechanisms, explainability approaches, and reported limitations. Quantitative findings were discussed only when the experimental setting and comparison conditions were sufficiently clear, and numerical results were not generalized across incompatible datasets or application domains.

3. Background

Fault diagnosis encompasses methodologies that aim to detect, isolate, and identify abnormal conditions in engineering systems to ensure reliability, safety, and operational efficiency. The diagnostic process typically includes fault detection (determining whether a fault has occurred), fault classification (identifying the type of fault), fault localization (pinpointing the faulty component), and, in advanced settings, fault prognosis or remaining useful life estimation. In modern cyber-physical systems, faults may arise from component degradation, sensor malfunctions, communication failures, or external disturbances, often exhibiting complex propagation patterns due to interdependencies among subsystems. These characteristics necessitate diagnostic approaches capable of modeling nonlinear relationships, dynamic interactions, and system-wide dependencies.

Traditional diagnostic methods can be broadly categorized into model-based, signal-based, and data-driven approaches. Model-based techniques rely on first-principles physics or analytical redundancy to generate residuals for fault detection. Although highly interpretable, they require accurate system models and become impractical for large-scale or evolving systems. Signal-based methods leverage time-frequency analysis, spectral decomposition, and statistical feature extraction to identify anomalies in sensor data and have been widely applied in rotating machinery and structural health monitoring. Data-driven approaches, including classical machine learning models such as support vector machines and random forests, improve diagnostic performance by learning patterns from historical data but often depend heavily on handcrafted features and struggle to capture complex system interactions. Deep learning methods, including CNNs for spatial feature extraction and RNNs or LSTMs for temporal modeling, address feature engineering limitations; however, their reliance on Euclidean data representations restricts their ability to model relational dependencies among distributed system components [6, 7]. Figure 3 provides a visual overview of the evolution of diagnostic approaches

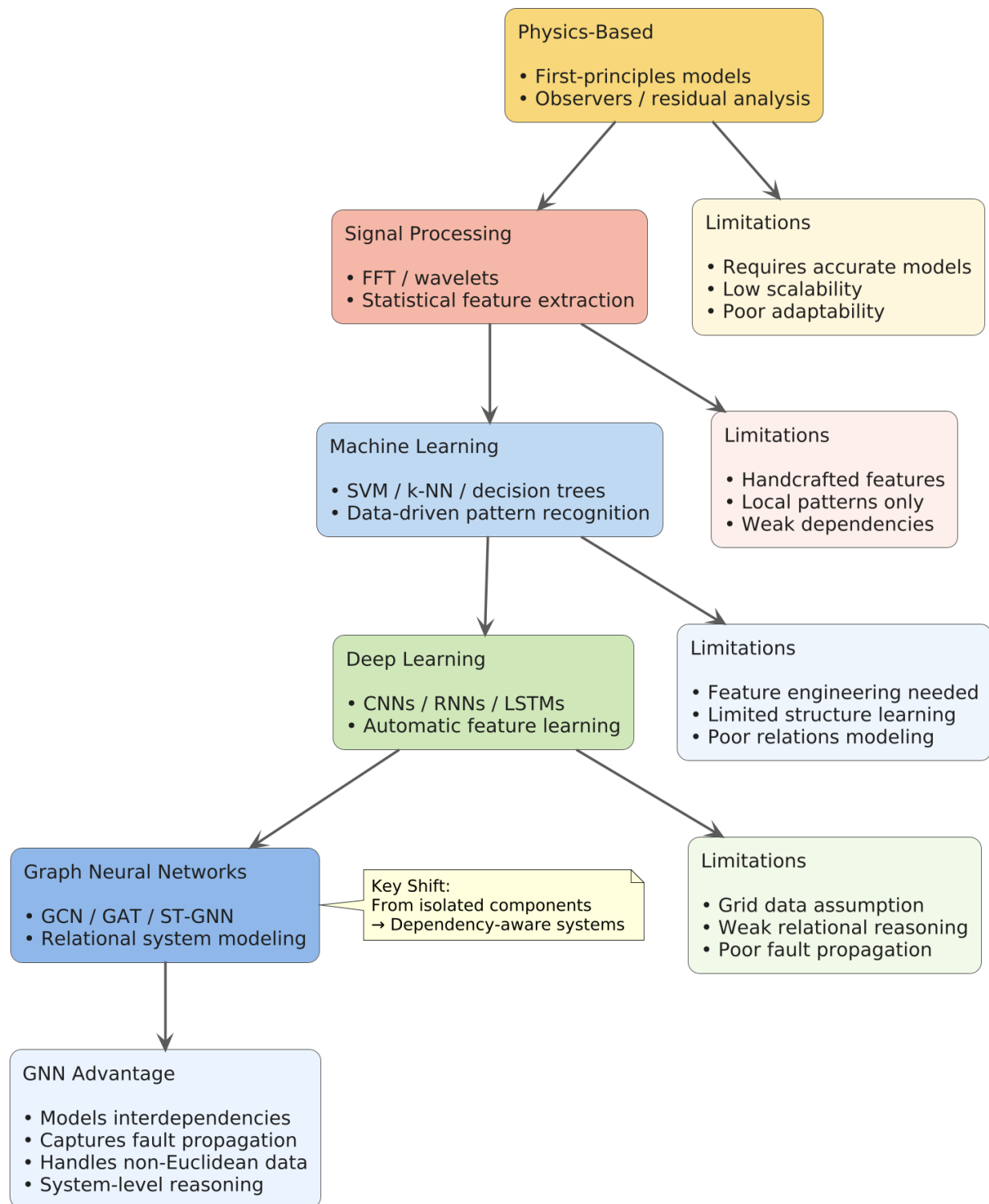


Figure 3: Evolution of fault diagnostic approaches, highlighting the limitations of prior methods and the transition to graph neural networks (GNNs) for relational, system-level modeling[7, 14].

and the key limitations that motivate the shift to graph neural networks.

Graph-based system modeling provides a useful abstraction for representing complex engineered systems, where nodes denote sensors, components, or subsystems, and edges encode physical connections, functional dependencies, or statistical correlations. This representation enables explicit modeling of fault propagation pathways and intercomponent influences that are difficult to capture using traditional grid-based learning methods. Graph

representations may be derived from physical topology, correlation analysis, or learned adaptively from data, allowing flexibility across diverse application domains. Within this paradigm, Graph Neural Networks (GNNs) extend deep learning to non-Euclidean domains through iterative message passing among neighboring nodes, thereby aggregating local and global structural information. Common architectures such as Graph Convolutional Networks (GCNs), Graph Attention Networks (GATs), and spatiotemporal GNNs model both structural dependencies and temporal

dynamics, making them well suited for fault diagnostics in interconnected, sensor-rich environments [7, 8].

Recent developments have extended GNN-based fault diagnosis through explainable and causal reasoning, adaptive graph construction, spatiotemporal modeling, and hypergraph representations. These approaches seek to improve interpretability, reduce sensitivity to spurious relationships, accommodate changing system structures, and represent higher-order dependencies in multivariate and multisensor environments [6, 8, 17, 18, 19]. Collectively, they establish the methodological foundation for the task-specific, deployment-oriented, and reliability-focused developments examined in the subsequent sections.

4. Historical Evolution of Fault Diagnostics Toward GNNs

Fault diagnostics has evolved in response to advances in sensing technologies, computational capabilities, and data availability. Early diagnostic systems were predominantly physics-based, relying on analytical models and expert knowledge to identify deviations from expected behavior. Observer-based methods, parity-space techniques, and analytical redundancy provided high interpretability and reliability in well-characterized systems but became difficult to apply in large-scale, nonlinear, or dynamically changing environments. As industrial systems became increasingly sensor-rich, signal-processing methods gained prominence, using spectral analysis, wavelet transforms, and statistical features to identify anomalies in vibration, acoustic, and electrical signals. Although effective for localized diagnosis, these approaches provided limited representation of interdependencies among system components [14]. Figure 4 illustrates the historical progression from physics-based and signal-processing approaches to data-driven and graph-based methods.

The emergence of data-driven methods marked a transition toward automated diagnostics. Classical machine learning techniques, including support vector machines, k-nearest neighbors, and decision trees, enabled pattern recognition from historical fault data but remained dependent on manual feature engineering and often struggled with high-dimensional, nonlinear relationships. Deep learning further reduced the need for handcrafted features by learning hierarchical representations directly from sensor data. Convolutional neural networks were applied to spatial and time-frequency patterns, while recurrent neural networks and long short-term memory models supported temporal analysis. However, these architectures generally assume Euclidean data structures and provide limited representation of relational dependencies and fault-propagation pathways in interconnected systems [7, 14]. Table 1 summarizes the strengths and limitations associated with each stage of this evolution.

The transition toward graph-based diagnostics arose from these limitations and from the inherently networked structure of modern cyber-physical systems. Early graph-based methods used graphical models and network analysis to represent component interactions and potential failure-propagation paths. Graph Neural Networks extended this principle by learning directly from structured system representations through message passing and neighborhood

aggregation [14]. Since the late 2010s, GNN-based methods have increasingly supported fault detection, classification, localization, and prognostics by combining local component information with system-level relational context.

Table 1: Evolution of fault diagnostic approaches

Era	Representative methods	Strengths	Limitations
Physics-based	Observers, parity-space methods, analytical redundancy	High interpretability; reliable in well-modeled systems	Requires accurate models; limited scalability in complex systems
Signal-based	FFT, wavelet transforms, statistical features	Effective for localized fault detection; established analytical methods	Limited ability to model system interactions
Classical machine learning	SVM, k-NN, decision trees	Data-driven pattern recognition; reduced dependence on physical models	Requires manual feature engineering; limited representation of nonlinear relationships
Deep learning	CNNs, RNNs, LSTMs	Automated feature extraction; temporal and spatial modeling	Assumes Euclidean data structures; limited relational modeling
Graph neural networks	GCN, GAT, ST-GNN	Models system dependencies; supports fault localization and graph-based analysis	Graph-construction sensitivity; computational overhead

From 2020 onward, research expanded beyond foundational convolutional, attention-based, and spatiotemporal GNNs toward more adaptive, interpretable, and deployment-oriented approaches. Recent developments include causal and explainable models, dynamic graph learning, semi-supervised and open-set methods, hypergraph representations, and hybrid Graph-Transformer architectures [6, 7, 8, 9, 17]. This progression reflects a broader shift from isolated component analysis toward system-level fault diagnosis under limited labels, evolving operating conditions, and practical deployment constraints.

5. Related Work

The application of Graph Neural Networks (GNNs) to fault diagnostics has expanded with the need to represent relational dependencies and fault-propagation dynamics in interconnected cyber-physical systems. Early studies adapted foundational architectures, including Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs), to model sensor correlations and structural topologies, while subsequent work introduced spatiotemporal extensions for evolving fault patterns. More recent studies have examined explainability, causal inference, adaptive graph learning, and multimodal fusion across industrial processes, renewable-energy systems, and marine machinery [20, 21, 22]. The literature primarily addresses four diagnostic objectives: fault detection, classification, localization, and prognostics. Although these tasks share graph-based modeling foundations, they differ in learning

Timeline of Fault Diagnostics Evolution Toward Graph Neural Networks

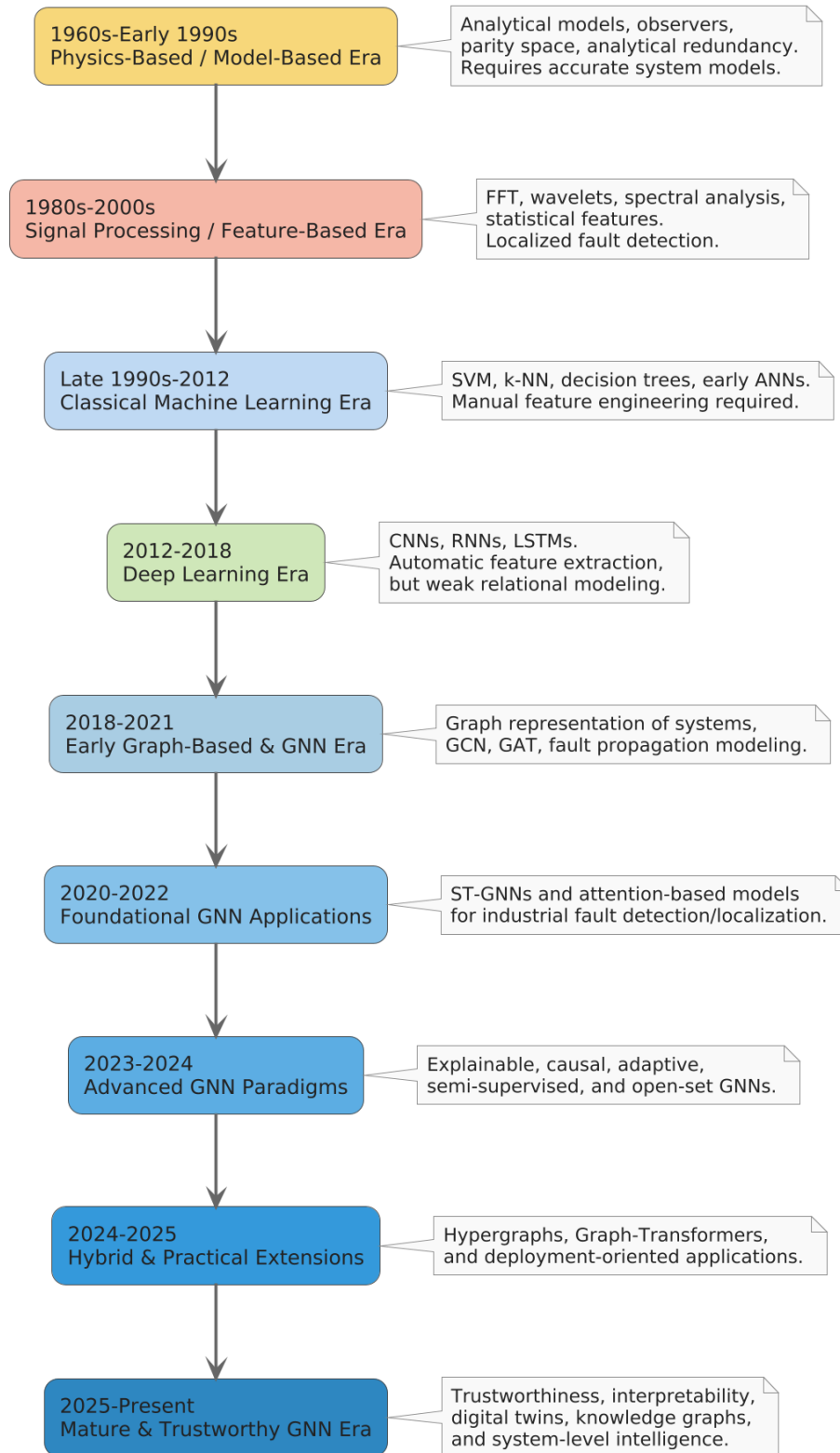


Figure 4: Historical evolution of fault diagnostic methodologies from physics-based and signal-processing approaches to classical machine learning, deep learning, and graph neural network (GNN)-based paradigms. The figure highlights the transition toward relational and system-level modeling in complex cyber-physical systems [6, 7, 9, 14, 17].

paradigms, data modalities, and evaluation requirements. This section reviews representative contributions, summarizes cross-cutting gaps, and compares the present survey with previous reviews.

5.1. Fault Detection

Fault detection identifies anomalous system states without necessarily assigning them to predefined fault categories and is commonly formulated as a graph anomaly-

detection problem. Early graph-based studies used graph convolution to combine measurements, component relationships, and engineering knowledge. In [23], the authors incorporated measurement information and prior knowledge into a graph convolutional diagnostic model, while in [24] authors modeled relationships among dissolved-gas measurements for power-transformer diagnosis. Spatiotemporal GNNs subsequently incorporated temporal dependencies, and attention-enhanced variants improved the weighting of spatial and temporal relationships in noisy sensor environments [25].

Recent work has explored self-supervised and contrastive learning to reduce dependence on labeled fault data. Pretext tasks such as edge masking and node augmentation support representation learning for anomaly scoring, while hierarchical spatiotemporal models capture interactions across multiple subsystems and scales [21]. Nevertheless, real-time detection and open-set recognition remain challenging when unseen fault signatures emerge under changing operating conditions. Current methods increasingly emphasize label efficiency and robustness, but validation in dynamic deployment environments remains limited [25].

5.2. Fault Classification

Fault classification assigns detected anomalies to specific fault categories and frequently uses graph representations for multisensor fusion. Foundational GCN-based studies demonstrated that structured measurement relationships and engineering knowledge could support classification across interconnected variables [23, 24]. Subsequent multiview approaches integrated time-domain, frequency-domain, and wavelet representations to capture complementary fault characteristics in heterogeneous sensor environments [20].

Recent research has focused on classification under noisy, imbalanced, and changing operating conditions. Noise-robust aggregation, adaptive graph weighting, and multiscale representations have been used to improve sensitivity to weak or minority-class fault patterns, while semi-supervised propagation incorporates unlabeled observations when annotated samples are limited. Despite these developments, changes in operating regimes and sensor configurations continue to reduce cross-domain performance [20]. Transferability across machines, facilities, and operating conditions therefore remains unresolved.

5.3. Fault Localization

Fault localization identifies the source or affected components of a fault by exploiting the relational structure of the system graph. In [26], the authors proposed a two-stage physics-informed framework that embedded power-grid geometry into a GNN and used physical similarity among labeled and unlabeled samples to support localization under sparse observations and limited labels. This work demonstrated how physical topology and graph learning can jointly support component-level fault identification.

Subsequent methods have used node, edge, and sub-graph importance to identify components associated with diagnostic outcomes. Adaptive spatiotemporal and hyper-graph models can additionally represent bidirectional and

higher-order dependencies relevant to fault propagation. However, localization performance remains sensitive to graph quality and structural mismatch. Inaccurate or static graph priors may propagate attribution errors, indicating the need for reliable and adaptive graph construction [25].

5.4. Prognostics and Remaining Useful Life Estimation

Prognostics and remaining useful life (RUL) estimation aim to predict degradation trajectories and expected failure horizons. In [27], the authors introduced a hierarchical attention graph convolutional network that represented sensors as graph nodes, modeled spatial dependencies through hierarchical graph convolution, and captured temporal behavior using bidirectional long short-term memory modeling. This study provided an early demonstration of graph-based multisensor prognostics.

Subsequent work incorporated gated graph convolution, temporal graph modeling, attention, and uncertainty-aware prediction. Spatiotemporal GNNs have been applied to uncertainty estimation, while Graph-Transformer hybrids have supported frequency-domain degradation analysis. Dynamic graph models can also represent changing inter-sensor relationships and previously unseen degradation modes. However, limited availability of long degradation sequences and weak generalization across operating conditions continue to restrict evaluation and deployment [22]. Current prognostic research therefore increasingly emphasizes adaptive and uncertainty-aware modeling, but still lacks sufficient long-horizon and cross-domain validation.

5.5. Cross-Cutting Challenges and Research Gaps

Across diagnostic tasks, recurring challenges include static graph assumptions, label scarcity, class imbalance, limited interpretability, and inconsistent evaluation practices. Adaptive graph learning and semi-supervised methods address some of these limitations, but their performance under real-time, resource-constrained, and changing operating conditions remains insufficiently validated. Differences in graph construction, preprocessing, data partitioning, and metric selection further hinder reproducibility and direct comparison across studies [28].

Additional gaps concern the physical validity of learned relationships and explanations, robustness to structural and distributional changes, and the integration of privacy-preserving, multimodal, and adversarially robust learning. These issues are examined in greater detail in the later sections on methodological limitations, operational failure modes, deployment architectures, and emerging research directions. Collectively, they motivate a survey framework that connects graph-learning methodology with benchmarking, reliability, and deployment requirements rather than treating these concerns independently.

5.6. Comparison with Existing Surveys and Novelty of This Review

Several reviews have examined GNN-based fault diagnosis, but most concentrate on a specific domain, diagnostic task, or methodological issue. In [14], the authors provided an early general review of GNN-based fault diagnosis and summarized foundational graph-learning formulations and representative applications. In [22], the

Table 2: Comparison of representative surveys related to GNN-based fault diagnosis

Survey	Primary scope	Multi-domain coverage	Unified taxonomy	Benchmark analysis	Deployment analysis	Operational failure modes	Ethics and safety	Practitioner guidance	Principal distinction
Study [14]	General GNN-based fault diagnosis	Partial	Partial	Limited	Not explicit	Not explicit	Not explicit	Not explicit	Early synthesis of graph-learning formulations and representative fault-diagnosis applications
Study [22]	GNN-based RUL prediction and prognostics	Partial	Task-specific	Primary	Limited	Not explicit	Limited	Not explicit	Focused analysis of GNN methodologies, evaluation practices, and future trends for RUL estimation
Study [15]	Process soft sensing, monitoring, and fault diagnosis	Domain-specific	Domain-specific	Limited	Limited	Not explicit	Limited	Not explicit	Industrial-process-centered review of spatiotemporal, attention-based, and hybrid GNN methods
Study [7]	Rolling-bearing fault diagnosis	Domain-specific	Domain-specific	Partial	Limited	Not explicit	Limited	Limited	Machinery-specific synthesis of graph construction and GNN methods for bearing diagnostics
Study [6]	Explainable GNNs for process fault diagnosis	Domain-specific	Explainability focused	Limited	Limited	Partial	Primary	Limited	Focused examination of explainability techniques and their role in process fault detection and diagnosis
Study [3]	Knowledge-graph-enhanced fault diagnosis and sensor management	Partial	Bibliometric	Not primary	Not explicit	Not explicit	Limited	Not explicit	Bibliometric analysis of publication growth, thematic trends, and knowledge-graph applications
Present survey	GNN-based fault diagnostics in cyber-physical systems	Primary	Primary	Primary	Primary	Primary	Primary	Primary	Integrated methodological, benchmark, deployment, operational, safety, and practitioner-oriented analysis

authors focused on GNN methodologies for remaining useful life prediction, including prognostic architectures, evaluation practices, and future trends. In [15], the authors reviewed process soft sensing, monitoring, and fault diagnosis, with emphasis on industrial-process applications. In [7] authors concentrated on rolling-bearing diagnosis and graph-construction strategies. In [6], the author examined explainable neural networks and GNNs for process fault diagnosis, while in [3] authors presented a bibliometric analysis of knowledge-graph-enhanced fault diagnosis and sensor management.

Table 2 compares the thematic coverage of these reviews with the present survey. Existing studies provide valuable analyses of GNN architectures, prognostics, process monitoring, rotating-machinery diagnosis, explainability, and knowledge-graph applications. However, benchmark reproducibility, deployment architectures, operational failure modes, ethical and safety concerns, and practitioner-oriented requirements are generally outside their principal scope. The present survey addresses this gap by connecting methodological developments with the reliability and deployment requirements of cyber-physical systems.

The novelty of this survey lies in its multidimensional, deployment-aware, and system-oriented synthesis. First, it introduces a unified taxonomy that jointly organizes the literature according to graph representation, learning paradigm, diagnostic objective, and deployment context, thereby showing how graph design, learning strategy, diagnostic purpose, and implementation environment jointly influence system behavior.

Second, the survey analyzes benchmark datasets and

evaluation practices while identifying reproducibility threats associated with inconsistent preprocessing, graph construction, data partitioning, and metric selection. Third, it examines cloud, edge, hybrid, federated, distributed, and digital-twin-supported architectures in relation to computational, communication, privacy, and latency constraints. Fourth, it distinguishes methodological limitations from operational failure modes, including cascading misdiagnosis, topology drift, noise amplification, open-set misclassification, adversarial vulnerability, oversmoothing, and concept drift. Fifth, it connects emerging technical directions with ethical, safety, governance, and practitioner-oriented considerations. The present survey therefore extends prior reviews by relating advances in GNN-based fault diagnosis to the operational reliability requirements of real-world cyber-physical systems.

6. Application Domains

Graph Neural Networks (GNNs) have been applied to fault diagnostics across industrial and cyber-physical systems because they can represent relational dependencies and fault-propagation mechanisms in interconnected infrastructures. Although many methods originate from general graph-learning principles, domain-specific system structures, sensing modalities, operating conditions, and safety requirements influence graph construction, feature representation, diagnostic objectives, and evaluation practices. This section reviews the principal application domains and identifies the characteristics that shape GNN-based diagnostic design.

6.1. Industrial Manufacturing Systems

Industrial manufacturing is a widely studied domain for GNN-based fault diagnostics, particularly for rotating machinery, rolling bearings, gearboxes, and robotic systems. Graphs are commonly constructed from multisensor vibration measurements, physical sensor layouts, component relationships, or correlation-based adjacency matrices. Early applications used graph convolutional networks to model intersensor dependencies and reported improvements in fault-classification performance over conventional neural networks under specific experimental conditions [29]. Spatiotemporal GNNs subsequently extended these approaches by representing interactions among components under varying loads and operating speeds.

Recent methods incorporate multiview features, adaptive graph weighting, and hybrid Graph–Transformer architectures to improve diagnosis under noisy and imbalanced conditions. Explainability mechanisms have also been used to support component-level fault attribution. Nevertheless, cross-condition generalization, real-time inference on resource-constrained devices, and adaptation to evolving operating regimes remain important challenges [29].

6.2. Power Systems and Smart Grids

Power systems possess an inherent graph structure in which buses, transmission lines, substations, and distributed energy resources form interconnected networks. GNNs have therefore been applied to fault detection and localization in distribution grids, photovoltaic-integrated systems, and microgrids by using physical topology as a graph prior. Early studies frequently relied on static network representations and reported improved localization performance over rule-based or statistical baselines in specific evaluation settings [25].

Dynamic and spatiotemporal graph models have subsequently been used to represent changing loads, renewable-energy variability, and network reconfiguration. Attention mechanisms can emphasize weak fault signatures and propagation pathways, while uncertainty-aware methods support reliability assessment in large networks. Major challenges include scalability to large grids, adaptation to changing topology, adversarial robustness, and real-time operation [25].

6.3. Transportation Systems

In transportation infrastructures, including railways, aviation systems, and autonomous vehicles, GNN-based diagnostic methods model relationships among sensors, mechanical components, infrastructure elements, and control subsystems. Graph representations may encode physical connectivity, functional dependencies, or temporal degradation relationships in rail tracks, aircraft engines, and vehicular networks [30].

Hybrid Graph–Transformer and heterogeneous graph architectures have been investigated to integrate frequency-domain features, multiple sensing modalities, and infrastructure data. Evolvable GNN frameworks additionally support incremental diagnosis as new fault categories emerge, reducing the need to train independent models for every condition [30]. However, heterogeneous sensing configurations, sparse failure records, safety-critical

validation requirements, and the absence of standardized benchmarks continue to limit cross-system evaluation and generalization.

Related applications include aerospace structural-health monitoring, pipeline and equipment diagnosis in oil and gas infrastructure, and proactive monitoring of servers, communication networks, and cooling systems in data centers. In each case, graph representations are used to capture dependencies that cannot be adequately modeled through isolated component analysis.

6.4. Cyber-Physical and IIoT Systems

Industrial Internet of Things (IIoT) environments contain distributed sensors, actuators, controllers, and communication devices whose interactions naturally form graph structures. GNN-based methods support anomaly detection by modeling cross-device dependencies, communication patterns, and subsystem interactions. Semi-supervised and open-set approaches are particularly relevant because labeled failures are limited and previously unseen fault types may arise during operation [21].

Federated and distributed GNNs have also been investigated for collaborative diagnosis across geographically separated facilities without centralizing raw data. Other approaches embed domain knowledge or temporal constraints to improve generalization under distribution shifts. Practical adoption remains constrained by communication overhead, computational limitations, nonstationary network conditions, and the need for timely inference in distributed environments [28, 21].

6.5. Healthcare and Biomedical Systems

Healthcare monitoring and biomedical diagnostics use GNNs to represent relationships among physiological signals, wearable sensors, anatomical structures, and clinical variables. Applications include anomaly detection in intensive-care monitoring, cardiac-signal analysis, and multisensor wearable systems. Graph-based models can integrate electronic health records, medical images, and sensor measurements while identifying physiological variables associated with abnormal conditions [31].

Multimodal and patient-specific graph architectures have also been applied to clinical decision support, chronic-disease management, drug-interaction analysis, and personalized monitoring. In this domain, diagnostic accuracy must be accompanied by interpretable outputs, privacy protection, regulatory compliance, and robustness across heterogeneous patient populations [31].

Table 3 summarizes the graph-construction strategies, diagnostic tasks, and principal challenges associated with these domains. Although graph structures range from physics-informed topologies to learned adjacency matrices, they share the objective of representing dependencies that influence fault occurrence and propagation. Domain-specific constraints related to scalability, sensing heterogeneity, privacy, interpretability, and data availability determine the suitability of particular GNN architectures and evaluation procedures.

Table 3: Representative application domains of GNN-based fault diagnostics

Domain	Graph-construction basis	Typical tasks	Key challenges
Industrial manufacturing	Sensor correlations, physical layouts, learned adjacencies	Detection, classification, localization	Noise sensitivity, class imbalance, cross-condition generalization
Power systems	Physical grid topology, dynamic load graphs	Fault localization, cascading-failure detection	Scalability, renewable-energy variability, adversarial robustness
Transportation	Component interactions, temporal degradation graphs	Fault detection, RUL estimation	Heterogeneous sensors, sparse failure data
IIoT systems	Device-communication graphs, subsystem dependencies	Anomaly detection, open-set recognition	Label scarcity, distributed-deployment constraints
Healthcare	Physiological interaction graphs, wearable-sensor networks	Anomaly detection, interpretable diagnostics	Regulatory compliance, patient variability, explainability

7. Unified Taxonomy of GNN-Based Fault Diagnostics

Existing literature frequently categorizes GNN-based fault diagnostics according to individual tasks, including detection, classification, localization, and prognostics. However, task-based categorization alone does not capture variation in graph representations, learning strategies, diagnostic objectives, and deployment environments. This section introduces a unified taxonomy across four complementary dimensions: graph structure and representation, learning paradigm, diagnostic objective, and deployment context. Figure 5 illustrates how these dimensions jointly define the design space of current approaches. The taxonomy supports systematic comparison, reveals relationships among methodological choices, and identifies underexplored combinations in the literature.

7.1. Graph Structure and Representation

A central source of variation in GNN-based fault diagnostics is how system interdependencies are encoded as graphs. Static graphs represent invariant physical or functional relationships through sensor layouts, correlation matrices, or prior engineering knowledge and are common in fixed-topology machinery and process systems [23]. Dynamic graphs represent temporal changes in relationships and are more suitable for varying loads, network reconfiguration, evolving degradation, and changing system topology [8, 26].

Heterogeneous graphs incorporate multiple node and edge types to represent interactions among sensors, actuators, components, physical flows, and information exchanges. Hypergraphs extend pairwise modeling by representing higher-order relationships among groups of components. For example, multi-metric fusion hypergraph networks construct hyperedges from instance, distributional, and spatiotemporal relationships in rotating-machinery diagnosis [16]. Adaptive and multirelational graph constructions have also been investigated to represent multiple

forms of mechanical and sensor dependence [19]. The selected representation influences fault-pathway modeling, noise sensitivity, computational requirements, and generalization across operating conditions.

7.2. Learning Paradigms

Learning paradigms vary according to data availability, labeling constraints, and operational requirements. Supervised learning remains common when sufficient labeled fault samples are available, particularly for classification and localization using graph convolutional and attention-based architectures [23]. Semi-supervised, self-supervised, and few-shot methods address label scarcity by learning from unlabeled or sparsely labeled graph data [21].

Open-set approaches extend conventional classification by identifying or rejecting fault categories absent during training. For example, a semi-supervised GCN-based framework uses labeled observations and pseudo-labeled samples to distinguish known and unknown faults in marine machinery [32]. Federated learning enables collaborative training across distributed facilities without centralizing raw operational data, while causal and physics-informed approaches incorporate domain knowledge or physical constraints to reduce dependence on spurious statistical relationships [26, 33]. Reinforcement learning represents an additional decision-oriented paradigm, although its application to GNN-based fault diagnosis remains comparatively limited.

7.3. Diagnostic Objectives

Diagnostic objectives define the functional scope of GNN-based systems. Fault detection identifies deviations from normal operation through anomaly scores, reconstruction losses, or temporal prediction errors. Fault classification assigns anomalies to predefined categories using relational feature aggregation and multisensor fusion [23]. Fault localization identifies affected components through

Unified Taxonomy of GNN-Based Fault Diagnostics

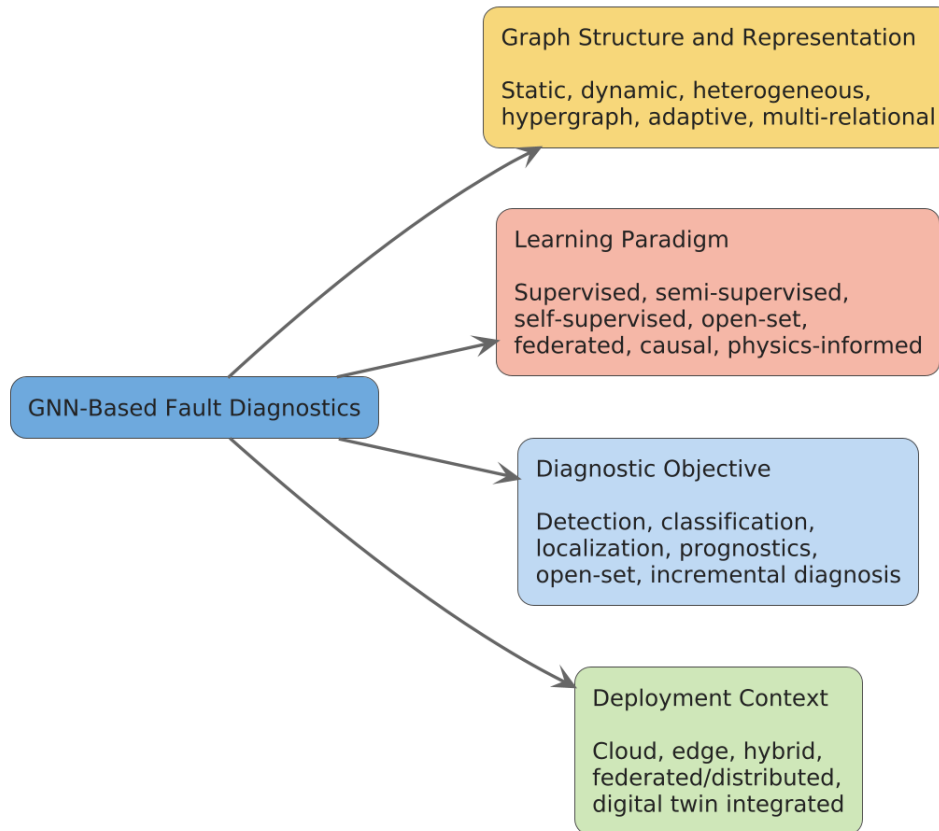


Figure 5: Unified multidimensional taxonomy of GNN-based fault diagnostics. Existing approaches are organized according to four complementary dimensions: graph structure and representation, learning paradigm, diagnostic objective, and deployment context. The taxonomy provides a structured basis for analyzing and comparing GNN-based diagnostic systems in complex cyber-physical environments.

graph topology, message passing, node-level importance, or physics-informed relationships [26].

Prognostics and remaining useful life estimation model degradation trajectories by combining intersensor dependencies with temporal information [27]. Open-set and incremental approaches extend these tasks to unknown or evolving fault modes [32]. Hierarchical spatiotemporal networks and adaptive graph aggregation have additionally been used for multiscale industrial-process diagnosis and weak-fault-feature detection [8, 21]. These objectives may be addressed independently or combined within integrated diagnostic and predictive-maintenance systems.

7.4. Deployment Context and System Architecture

Deployment context determines where graph construction, model training, inference, and updating are performed and therefore influences latency, computational demand, bandwidth use, and privacy. Cloud-centric architectures support computationally intensive training and large-scale graph processing but may introduce communication delay, bandwidth dependence, and exposure of operational data. Edge-oriented architectures perform inference near sensors or equipment, enabling rapid responses while requiring lightweight models, reduced memory use, efficient graph sampling, and optimized message passing.

Hybrid edge-cloud architectures divide these responsibilities by performing time-sensitive detection locally and using cloud resources for model retraining, aggregation, or

long-term analysis. Federated and distributed architectures enable collaborative learning across facilities or network domains without transferring raw data to a central repository. Federated temporal GNNs, for example, have been evaluated for fault detection in operational telecommunication networks [33]. Such deployments nevertheless face communication overhead, non-independent and identically distributed data, heterogeneous computational resources, synchronization requirements, and changes in local graph topology. These operational trade-offs are examined in greater detail in the dedicated deployment-architecture section.

The multidimensional taxonomy clarifies relationships among graph representation, learning paradigm, diagnostic objective, and deployment context. It also reveals underexplored combinations, such as federated dynamic heterogeneous GNNs for open-set prognostics, physics-informed hypergraphs for causal fault localization, and explainable semi-supervised models for edge-deployed renewable-energy systems. These combinations represent prospective directions rather than established methodological categories. Overall, the taxonomy provides a basis for comparative evaluation and standardized reporting while emphasizing that graph design, learning efficiency, diagnostic granularity, and deployment feasibility must be considered jointly when developing reliable GNN-based fault-diagnostics systems.

8. Benchmark Datasets and Evaluation Protocols

The lack of standardized benchmarks and consistent evaluation protocols continues to limit reproducibility and direct comparison in Graph Neural Network (GNN)-based fault diagnostics. Existing studies frequently use domain-specific datasets that differ in fault scenarios, sensor configurations, sampling frequencies, operating conditions, preprocessing procedures, and graph-construction strategies. These differences complicate methodological comparison and the assessment of generalization across systems. Shared benchmark suites, consistent preprocessing pipelines, and transparent reporting practices are therefore needed to support comparable evaluation and industrial validation [34, 35]. Benchmark design should also reflect the diversity of graph representations, learning paradigms, diagnostic objectives, and deployment settings identified in the unified taxonomy.

8.1. Widely Used and Emerging Benchmark Datasets

Several publicly available datasets are commonly used as benchmarks across fault-diagnostic domains, although differences in data structure and experimental protocols limit cross-study comparison. In rotating-machinery diagnostics, the Case Western Reserve University (CWRU) bearing dataset is widely used for evaluating fault-classification methods. It contains vibration measurements for inner-race, outer-race, and ball faults under different motor loads and operating speeds. Drive-end bearing measurements are available at sampling rates of 12 kHz and 48 kHz, while fan-end measurements are provided at 12 kHz [36]. Table 4 compares representative benchmark datasets used in GNN-based fault diagnostics according to their application domain, diagnostic objective, data characteristics, typical graph-construction strategy, commonly reported evaluation metrics, and principal limitations. The graph-construction column describes representative strategies adopted in the reviewed GNN literature rather than graph structures inherently supplied by every dataset. This distinction is important because most conventional fault-diagnostic datasets provide multivariate signals but do not prescribe a unique graph representation; consequently, differences in node definition, edge construction, preprocessing, and train-test partitioning can substantially affect the reported performance.

Other bearing datasets include the Paderborn University benchmark, which provides synchronously measured vibration and motor-current signals for healthy and damaged bearings under several operating conditions [37]. The Intelligent Maintenance Systems (IMS) bearing dataset contains run-to-failure bearing experiments and is commonly used for degradation analysis and prognostics [38, 39]. The XJTU-SY dataset similarly provides complete run-to-failure vibration data from accelerated life tests involving multiple rolling-element bearings and operating conditions [40, 41]. These datasets support investigations of fault classification, domain adaptation, degradation modeling, and remaining useful life estimation.

Beyond vibration-based diagnosis, the MIMII dataset provides normal and anomalous sound recordings from industrial machines, including valves, pumps, fans, and slide rails, and supports acoustic anomaly-detection research

[42]. The MIMII DUE extension introduces operational and environmental domain shifts for evaluating robustness under changing conditions [43]. For aircraft-engine prognostics, the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dataset provides multiple run-to-failure time series generated under different operating conditions and fault modes [44, 45].

For industrial process monitoring, the Tennessee Eastman Process (TEP) is a widely used simulated chemical-process benchmark. The original process model defines 41 measured variables, 11 manipulated variables, and 21 predefined disturbance or fault conditions, supporting fault detection, classification, localization, and process-control studies [46]. Because modified and extended versions of the TEP dataset are also available, studies should identify the specific simulator version, selected variables, and fault configuration used.

In power-system research, standard IEEE bus and feeder test systems provide predefined network topologies for power-flow analysis, fault localization, cascading-failure analysis, and renewable-integration studies. PowerGraph extends this benchmark setting by providing GNN-oriented datasets for power flow, optimal power flow, and cascading-failure analysis. It supports both node-level and graph-level tasks and includes reference explanations for cascading-failure scenarios [35]. SafePowerGraph further introduces safety-oriented evaluation under conditions such as energy-price changes and transmission-line outages [47].

Additional resources include naval-propulsion simulation data, lithium-ion battery degradation datasets, and multimodal industrial datasets containing combinations of vibration, acoustic, thermal, electrical, and visual measurements. Despite the availability of these resources, reproducibility remains limited by inconsistent dataset versions, preprocessing choices, train-test splits, graph-construction procedures, and fault-severity definitions. Studies should therefore report sufficient information to reconstruct both the input signals and the graph representation supplied to the model.

8.2. Evaluation Metrics and Protocols

Evaluation metrics should be selected according to the diagnostic objective. Fault-classification studies commonly report accuracy, precision, recall, macro- and micro-averaged F1-scores, and confusion matrices. For imbalanced fault distributions, balanced accuracy, macro-F1, area under the receiver operating characteristic curve (AUC-ROC), and area under the precision-recall curve provide more informative evaluation than accuracy alone.

Fault-detection studies may report false-alarm rate, missed-detection rate, detection delay, early-detection latency, and reconstruction or prediction error. Fault-localization performance can be evaluated using top- k accuracy, mean localization error, node-level precision and recall, mean reciprocal rank, or graph distance between predicted and actual fault locations. Prognostic studies typically use root mean squared error (RMSE), mean absolute error (MAE), prognostic-horizon measures, and task-specific scoring functions for remaining useful life estimation.

Evaluation under a single random train-test split pro-

Table 4: Comparison of representative benchmark datasets used in GNN-based fault diagnostics

Dataset	Domain	Typical diagnostic task	Data characteristics	Typical graph construction	Common evaluation metrics	Principal limitations
CWRU [36]	Rotating machinery and bearing diagnosis	Fault detection and multiclass fault classification	Vibration signals representing healthy bearings and inner-race, outer-race, and ball faults under multiple motor loads and speeds; drive-end measurements at 12 and 48 kHz and fan-end measurements at 12 kHz	Nodes represent sensors, signal segments, frequency bands, or extracted features; edges are commonly defined using physical proximity, correlation, similarity, or learned adjacency	Accuracy, precision, recall, macro-F1, confusion matrix, AUC-ROC	Laboratory-induced faults; limited operating variability; possible data leakage from overlapping windows; results are highly sensitive to preprocessing and train-test splitting
Paderborn [37]	Rotating machinery and bearing diagnosis	Fault classification, domain adaptation, and condition-transfer evaluation	Synchronous vibration and motor-current measurements from healthy and damaged bearings under several operating conditions; includes artificial and naturally damaged bearings	Multimodal sensor or feature nodes connected through physical relationships, statistical dependence, feature similarity, or learned cross-modal edges	Accuracy, macro-F1, balanced accuracy, precision, recall, confusion matrix	Variation in experimental subsets and fault groupings; heterogeneous operating conditions complicate direct comparison; preprocessing and split protocols are not standardized
IMS [38, 39]	Bearing degradation and prognostics	Anomaly detection, degradation-stage identification, and remaining useful life estimation	Run-to-failure vibration measurements collected from bearings operated until failure	Nodes represent bearings, sensors, temporal windows, or degradation states; edges encode sensor relationships, temporal continuity, or similarity between operating states	RMSE, MAE, prognostic score, detection delay, early-warning time	Small number of run-to-failure trajectories; limited diversity of operating conditions; uncertain failure-onset labeling; restricted support for cross-domain generalization
XJTU-SY [40, 41]	Bearing degradation and accelerated-life testing	Health-state assessment, degradation modeling, and remaining useful life prediction	Complete run-to-failure vibration signals from multiple rolling-element bearings under several operating conditions	Sensor or temporal-window nodes connected using temporal adjacency, feature similarity, correlation, or dynamically learned degradation relationships	RMSE, MAE, MAPE, prognostic score, monotonicity, trendability	Accelerated laboratory degradation may not fully represent field failures; relatively limited operating regimes; model performance depends strongly on health-index construction and failure-threshold definition
MIMII /MIMII DUE [42, 43]	Industrial acoustic monitoring	Unsupervised or semi-supervised acoustic anomaly detection and domain-shift evaluation	Normal and anomalous recordings from valves, pumps, fans, and slide rails; MIMII DUE introduces changes in operating and environmental conditions	Nodes represent machines, microphones, time-frequency patches, or acoustic embeddings; edges are based on spatial relationships, spectral similarity, temporal association, or learned adjacency	AUC-ROC, partial AUC, F1-score, precision, recall, false-positive rate	Anomalies are collected under controlled conditions; strong sensitivity to background noise and domain shift; graph definitions vary substantially across studies
NASA MAPSS [44, 45]	C-Aircraft engine prognostics	Degradation modeling and remaining useful life estimation	Multivariate run-to-failure time series generated for turbofan engines under different operating conditions and fault modes	Nodes represent sensors, engine subsystems, temporal segments, or operating variables; edges are derived from physical knowledge, correlation, mutual information, attention, or learned adjacency	RMSE, MAE, NASA scoring function, MAPE, prediction-horizon error	Simulation-based data; sensor and subsystem relationships are not explicitly provided; operating-condition normalization and RUL-label construction differ across studies
TEP [46]	Chemical-process monitoring	Fault detection, classification, localization, and process anomaly diagnosis	Simulated chemical process with 41 measured variables, 11 manipulated variables, and 21 predefined disturbances or fault conditions	Nodes correspond to process variables, sensors, control loops, or units; edges are constructed from process topology, causal relations, correlation, mutual information, or learned dependency matrices	Accuracy, macro-F1, detection rate, false-alarm rate, detection delay, localization accuracy	Multiple simulator versions and modified fault sets are used; some faults are difficult to distinguish; controlled simulation does not fully capture industrial uncertainty and maintenance effects
PowerGraph [35]	Power-system graph learning	Power flow, optimal power flow, cascading-failure analysis, node-level and graph-level prediction, and explanation evaluation	GNN-oriented power-system benchmark containing graph-structured tasks and reference explanations for cascading-failure scenarios	Native electrical topology is used directly; nodes represent buses or power-system components and edges represent electrical connections with associated attributes	MAE, RMSE, classification accuracy, F1-score, explanation fidelity, sparsity, and localization agreement	Primarily simulation-based; performance may depend on network size and operating scenarios; limited evidence of transfer to real utility data and evolving topologies
Safe PowerGraph [47]	Safety-oriented power-system analysis	Robustness, safety assessment, and fault or disturbance prediction under operational changes	Graph-structured power-system scenarios incorporating disturbances such as energy-price variation and transmission-line outages	Physical grid topology with node and edge attributes reflecting system states, operational conditions, and disturbance scenarios	Accuracy, F1-score, MAE, RMSE, calibration error, robustness degradation, and safety-violation rate	Emerging benchmark with limited adoption; scenario coverage may not represent all cyber-physical threats; comparability with conventional fault-diagnosis datasets remains limited

vides limited evidence of operational reliability. Benchmarking studies therefore recommend consistent data splits, multiple random seeds, common baseline implementations, and comparable model-training procedures [34]. Robustness protocols should additionally examine signal

noise, missing sensors, topology perturbations, operating-condition changes, low-sample settings, class imbalance, and open-set faults. Cross-condition, cross-machine, and cross-domain evaluations can provide further evidence of generalization.

Open-set studies should report known-class performance together with unknown-fault rejection accuracy, false-unknown rate, and threshold-dependent novelty-detection metrics. For graph models intended for dynamic environments, evaluation should also quantify performance degradation under node removal, edge perturbation, topology reconfiguration, and temporal distribution shift. Safety-oriented power-system benchmarks similarly indicate the importance of testing GNNs under line outages and changing operating conditions rather than relying only on nominal test cases [47].

Uncertainty and explainability evaluations are also relevant for safety-critical applications. Uncertainty assessment may include expected calibration error, negative log-likelihood, Brier score, prediction intervals, or selective-risk analysis. Explainability assessment may consider fidelity, sparsity, stability, localization agreement, prototype consistency, and agreement with physical or causal knowledge. However, explanation compactness or stability alone does not establish physical validity, and explanations should therefore be evaluated against domain knowledge or known fault-propagation structures where possible.

8.3. Challenges in Benchmark Standardization

Benchmark standardization is constrained by several factors. Proprietary industrial datasets often cannot be released because of confidentiality, security, or commercial restrictions. Public datasets frequently contain controlled or artificially induced fault conditions that do not fully represent gradual degradation, simultaneous faults, maintenance interventions, sensor replacement, or changing operational regimes.

Preprocessing also varies substantially across studies. Differences in normalization, filtering, segmentation, window length, overlap, feature extraction, augmentation, and graph generation can affect reported performance. In addition, splitting overlapping windows from the same operating sequence across training and test sets can introduce information leakage and produce overly optimistic results.

Graph-specific design choices introduce further variation. Adjacency matrices may be derived from physical topology, correlation, distance, mutual information, learned relationships, or combinations of these sources. Studies also differ in their treatment of edge direction, edge weights, temporal updates, missing nodes, and topology changes. Without consistent reporting, it is difficult to determine whether an observed improvement results from the GNN architecture, graph construction, preprocessing, or the selected data split.

Dynamic topologies, temporal dependencies, multi-modal inputs, open-set faults, uncertainty, and adversarial robustness remain insufficiently represented in many current benchmarks. PowerGraph and SafePowerGraph illustrate recent efforts to extend benchmark design beyond conventional prediction accuracy toward graph-level tasks, explanations, operational disturbances, and safety-oriented evaluation [35, 47].

8.4. Toward Standardized Evaluation Frameworks

Future benchmark development should emphasize shared repositories, documented preprocessing pipelines,

reference graph-construction strategies, reproducible baseline implementations, and predefined evaluation splits. General GNN benchmarking work has shown the value of common experimental protocols, standardized model implementations, and repeated evaluation across datasets [34]. Domain-specific resources such as PowerGraph similarly demonstrate how multiple power-system tasks and explainability analyses can be supported within a common graph-learning framework [35].

Standardized fault-diagnostic benchmarks should include static and dynamic graph variants, multiple operating conditions, class-imbalance scenarios, missing-sensor experiments, open-set evaluation, and cross-domain transfer. Robustness tests should cover signal noise, adversarial perturbations, sensor failures, topology changes, and communication constraints. Benchmark suites should also report computational measures, including training time, inference latency, memory consumption, parameter count, communication cost, and energy consumption where deployment efficiency is relevant.

Transparent reporting should specify dataset versions, sensor channels, sampling rates, windowing procedures, graph definitions, train-validation-test splits, random seeds, hyperparameter-selection procedures, and hardware configurations. Shared codebases and reproducible experiment configurations would support more reliable comparisons and provide clearer evidence regarding the suitability of GNN-based diagnostic methods for operational deployment.

9. Deployment Architectures for GNN-Based Fault Diagnostics

Although research has improved the predictive performance and interpretability of Graph Neural Network (GNN)-based fault-diagnostic models, their deployment in cyber-physical and industrial systems requires consideration of latency, privacy, scalability, communication overhead, computational resources, and integration with existing monitoring infrastructure. This section examines five deployment paradigms: cloud-centric, edge-oriented, hybrid edge-cloud, federated or distributed, and digital-twin-integrated architectures. These deployment contexts complement the unified taxonomy by showing how system architecture influences graph construction, learning strategy, inference location, and diagnostic response.

9.1. Cloud-Centric Deployment

Cloud-centric architectures are suitable for computationally intensive activities such as initial model training, large-scale graph processing, historical-data analysis, periodic model updating, and aggregation of data from multiple facilities. Centralized platforms provide the computing and storage capacity required for complex graph models and may also support simulation-based validation and digital-twin services. However, cloud-based inference requires operational data or intermediate representations to be transmitted over communication networks, which may introduce latency, bandwidth demands, dependence on network availability, and privacy concerns. Consequently, cloud deployment is generally more appropriate for offline analysis, global model management, and long-term

optimization than for time-critical local fault response in safety-critical environments.

9.2. Edge Deployment for Real-Time Diagnostics

Edge-oriented deployment performs GNN inference close to the monitored equipment through embedded processors, industrial gateways, local servers, or other edge devices, thereby reducing communication dependence and supporting faster diagnostic responses. The memory, computational, and energy requirements of GNN inference remain important constraints on resource-limited hardware, motivating methods such as graph sampling, model pruning, quantization, knowledge distillation, workload partitioning, and hardware-aware architecture search. GNN-specific quantization techniques have been developed to reduce inference complexity and memory consumption while limiting accuracy loss [48], while device-edge co-inference frameworks consider latency, memory limits, and communication cost during model design and deployment [49]. Physics-informed GNNs provide an additional mechanism for incorporating operational constraints into graph learning, as illustrated by GraPhyR, which integrates network connectivity and physical constraints for dynamic power-system reconfiguration [50]. Despite these developments, edge-deployed models must still address changing graph structures, sensor failures, model drift, distribution shifts, and potential accuracy degradation caused by compression.

9.3. Hybrid Edge-Cloud Architectures

Hybrid edge-cloud architectures divide diagnostic processing between local and centralized resources by allowing edge nodes to perform data filtering, graph construction, anomaly detection, or preliminary fault classification, while cloud services handle model retraining, global aggregation, long-term analysis, and computationally intensive verification. Instead of transmitting complete sensor streams, edge devices may send compressed features, graph embeddings, anomaly scores, or selected subgraphs to reduce communication volume while retaining access to centralized computing resources. Research on device-edge GNN co-inference indicates that workload partitioning must jointly consider graph structure, device capacity, and communication cost because conventional layer-based partitioning may not be efficient for graph operations [49]. Hybrid deployment therefore offers a trade-off between local responsiveness and centralized analytical capacity, but its effectiveness depends on workload allocation, synchronization, communication reliability, fault tolerance, and the frequency and security of model updates.

9.4. Federated and Distributed Deployment

Federated learning enables multiple sites to train local GNN models without transferring raw operational data to a central repository, with a coordinating server aggregating model parameters or updates instead. This approach can support privacy, data-sovereignty, and regulatory requirements when industrial facilities or network operators cannot share complete sensor records. Bourgerie and Zanouda proposed a bi-level federated temporal GNN

for anomaly detection and diagnosis in telecommunication networks, representing both interactions among radio-access-network nodes and software-execution relationships within individual nodes while reducing raw-data sharing and communication costs [33]. Nevertheless, federated GNN deployment remains affected by non-independent and identically distributed data, graph heterogeneity, communication overhead, partial client participation, convergence instability, and vulnerability to unreliable or malicious participants. These limitations motivate further research on communication-efficient training, update validation, secure aggregation, and robust learning across heterogeneous graph structures.

9.5. Integration with Digital Twins and Real-Time Monitoring Pipelines

Digital twins provide virtual representations of physical systems that are continuously or periodically updated using operational data and can support simulation, synthetic-data generation, scenario analysis, predictive maintenance, and comparison between predicted and observed behavior. GNNs are well suited to digital-twin environments because physical, communication, and functional relationships can be represented as graph structures. Isah integrated graph Fourier transformation with a message-passing neural network for failure classification in real and simulated network-digital-twin environments, illustrating how graph learning can represent dependencies among network components and support failure analysis [51]. However, digital-twin-supported diagnosis requires reliable synchronization between physical and virtual systems, and its effectiveness may be reduced by delayed sensor updates, simulation-to-reality differences, model drift, incomplete physical models, and unreported uncertainty. Digital-twin pipelines should therefore document synchronization frequency, graph-update procedures, uncertainty propagation, and validation against physical-system observations.

9.6. Deployment Trade-Offs and Future Directions

Each deployment architecture involves distinct trade-offs. Cloud-centric systems provide substantial computational capacity but depend on network connectivity and centralized data management; edge systems support local inference but operate under tighter memory, energy, and processing constraints; hybrid architectures distribute these responsibilities but require synchronization and workload orchestration; and federated systems reduce centralized raw-data collection but introduce communication, optimization, and security challenges. Future research should evaluate deployment architectures using both predictive and operational measures, including inference latency, communication volume, memory consumption, model size, energy use, update frequency, fault-response time, and performance under topology or distribution shifts. Deployment-aware research should also examine physics-informed lightweight GNNs, communication-efficient federated learning, uncertainty-aware inference, adaptive edge processing, and reproducible edge-cloud pipelines to support the transition from experimental models to operational fault-diagnostic systems.

10. Limitations

Despite progress in Graph Neural Networks (GNNs) for fault diagnostics, several methodological and operational limitations continue to affect generalization, scalability, interpretability, and deployment in cyber-physical and industrial systems. Current research identifies recurring concerns related to graph-construction uncertainty, limited and imbalanced fault data, computational requirements, explanation validity, and robustness under domain shifts. Addressing these limitations is necessary for translating GNN-based diagnostic methods from controlled experimental settings to operational applications.

10.1. Graph Construction and Topology Uncertainty

Graph construction remains a central limitation because many approaches depend on predefined physical topologies, correlation-derived edges, or learned graph structures that may be incomplete, noisy, or subject to change because of system reconfiguration, sensor failures, maintenance actions, or the integration of distributed energy resources. Inaccurate or outdated graph representations can reduce diagnostic performance by misrepresenting intercomponent dependencies and fault-propagation pathways. Although adaptive graph learning can account for changing relationships, it may introduce additional computational cost and training instability. Photovoltaic-integrated distribution networks present further difficulties because variable renewable generation, sparse measurements, changing fault impedance, and low observation rates can produce weak fault signatures and uncertain graph relationships [25]. Similarly, rolling-bearing diagnosis under nonstationary operating conditions requires spatiotemporal graph representations that can account for changes in speed, load, and sensor relationships [29].

10.2. Data Scarcity, Class Imbalance, and Label Limitations

Industrial fault datasets are often limited, highly imbalanced, and dominated by samples representing normal operation, while rare, compound, or previously unseen faults remain underrepresented. Semi-supervised, self-supervised, few-shot, and open-set GNN approaches partially address limited labeling, but their effectiveness varies across datasets and application domains. Synthetic fault generation may increase sample availability, although simulated data may not reproduce realistic degradation processes, component interactions, or fault-propagation patterns [52]. Severe class imbalance can also bias decision boundaries toward normal or majority fault categories and reduce node- or graph-level performance for rare events. Mechanical systems and power networks continue to face limitations associated with small training sets and imbalanced labels, while process-industry applications often depend on high-quality labeled data and sufficient domain knowledge for model development and validation [21].

10.3. Scalability and Computational Efficiency

Message passing over large or densely connected graphs can require substantial memory, computation, and communication, particularly in large power grids, distributed Industrial Internet of Things environments, and applications

involving high-dimensional multisource sensor streams [21]. Increasing the number of nodes, edges, temporal windows, or graph layers can also increase inference latency and make deployment on resource-constrained devices more difficult. Lightweight architectures, neighborhood sampling, graph sparsification, pruning, quantization, and distributed processing can reduce computational requirements, but they may also remove diagnostically relevant relationships or reduce predictive accuracy. Evaluations in distribution-grid settings further show that topology changes can reduce model performance, indicating that computational efficiency alone does not ensure robustness to structural variation [11]. Consequently, scalability should be assessed together with latency, memory use, communication cost, graph size, and performance under topology changes.

10.4. Interpretability, Trustworthiness, and Causal Validity

Explainable GNN methods, including attribution techniques, prototype-based reasoning, and subgraph explanations, can provide information about the nodes, edges, or features associated with a diagnostic decision. However, many explanations are generated after prediction and may not reflect causal mechanisms or physically meaningful fault pathways. In safety-critical systems, visually plausible but causally invalid explanations can reduce confidence in diagnostic outputs and may reinforce spurious correlations or predictive shortcuts. Reviews of explainable graph learning note that different post-hoc methods can produce multiple plausible explanations for the same prediction, even when some explanations conflict with domain knowledge [6]. Causal intervention-based GNNs seek to reduce reliance on confounding relationships, but they introduce challenges involving causal-graph specification, variable coupling, training stability, and computational complexity [17]. Self-interpretable models may improve transparency, although restricting the model to interpretable structures can reduce representational flexibility in complex process-fault scenarios.

10.5. Generalization, Domain Shift, and Robustness

GNN-based diagnostic models may not generalize reliably across changes in operating conditions, system topology, sensor configuration, noise level, or fault characteristics. Models trained on one machine, facility, or network configuration may experience performance degradation when applied to another domain because both feature distributions and graph structures can change. Domain adaptation and transfer learning for graph-based fault diagnosis remain less developed than conventional supervised approaches, particularly for evolving power grids and systems containing distributed energy resources. Attention-based variants such as RGATv2 have reported lower F1-score degradation under selected topology changes than the evaluated baselines; however, such results remain dependent on the dataset, perturbation protocol, and comparison models and should not be interpreted as evidence of general robustness [52]. Previously unseen topologies, noisy observations, out-of-distribution faults, and simultaneous distribution shifts therefore remain open concerns, motivating further research on causal representation learning, uncertainty esti-

mation, domain adaptation, and evaluation across multiple operational settings [22].

Improving robustness may introduce trade-offs with predictive performance and deployment efficiency. Robust aggregation, adversarial training, uncertainty estimation, and dynamic graph adaptation can increase resilience to noisy observations, topology changes, and distribution shifts, but may reduce performance under clean or familiar operating conditions when regularization suppresses informative task-specific patterns. These mechanisms can also increase model complexity, training time, inference latency, and memory consumption, thereby limiting their suitability for real-time or edge-constrained diagnostic systems [22]. Attention-based and Graph-Transformer architectures can improve long-range dependency modeling and adaptation to graph variation, but their computational cost generally increases with graph size. Similarly, federated and privacy-preserving approaches reduce raw-data exposure while introducing communication overhead, convergence instability, and possible performance degradation under heterogeneous local data [53]. Robustness should therefore be evaluated jointly with clean-condition accuracy, calibration, latency, computational requirements, and resource consumption rather than treated as an isolated objective.

Collectively, these limitations indicate the need for diagnostic frameworks that combine physics-informed graph construction, causal reasoning, computationally efficient architectures, uncertainty estimation, and evaluation under realistic operating constraints. Future work should place greater emphasis on graph-quality assessment, cross-domain validation, rare-fault evaluation, computational reporting, and physically meaningful explanations to support reliable deployment of GNN-based fault-diagnostic systems.

11. Failure Modes of GNN-Based Fault Diagnostics

Although Graph Neural Network (GNN)-based fault-diagnostic systems often report strong performance on controlled benchmarks, their operation in cyber-physical and industrial environments may be affected by failure modes that are distinct from the methodological limitations discussed in Section 10. These failures can appear as incorrect diagnoses, delayed responses, inappropriate fault isolation, or unsafe downstream control decisions. Studies of dynamic topologies, noisy measurements, open-set conditions, adversarial manipulation, and changing operating environments indicate that benchmark performance alone does not establish operational reliability. Understanding these failure mechanisms is therefore important for designing GNN-based diagnostic systems for industrial, power, and transportation applications.

11.1. Cascading Misdiagnosis and Error Propagation

Message passing enables GNNs to aggregate information across related system components, but it can also propagate errors originating from sensor noise, calibration drift, missing measurements, or biased predictions. A local error may influence neighboring node representations across multiple message-passing layers and lead to incorrect estimates of fault location, severity, or root cause [11, 54]. In power-distribution networks, an erroneous

node classification may affect downstream fault-isolation decisions, potentially causing unnecessary feeder disconnection or delayed restoration. Similar propagation effects in industrial processes may produce incorrect root-cause attribution and inefficient maintenance actions. These risks motivate robust aggregation, uncertainty-aware message passing, confidence-based propagation control, and mechanisms for containing unreliable node information [10, 54, 55].

11.2. Topology Drift and Graph Mismatch

Many GNN-based diagnostic models are developed using fixed or slowly changing graph structures, whereas operational systems may undergo topology changes because of maintenance, component outages, line switching, distributed-energy-resource integration, equipment replacement, or changing IIoT connectivity. When the assumed graph differs from the actual system structure, message passing may occur across missing, outdated, or spurious edges, reducing fault-detection and localization performance. Evaluations on distribution-network topologies indicate that conventional GCN models can experience substantial performance degradation under structural perturbations, while relation-aware attention models may be more resistant under selected experimental conditions [11]. However, the degree of robustness depends on the perturbation type, dataset, and graph-construction procedure. Persistent graph mismatch is particularly relevant in renewable-integrated grids and reconfigurable manufacturing systems, where adaptive graph reconstruction and online topology estimation may be required to maintain diagnostic accuracy [11].

11.3. Noise Amplification and Sensor-Fault Propagation

Graph aggregation may spread localized measurement noise or corrupted sensor values to neighboring node embeddings when robust filtering, reliability weighting, or outlier rejection is absent [10, 20, 56]. Repeated propagation can increase false alarms, distort fault localization, or suppress weak fault signatures, particularly around highly connected nodes and in densely linked multisensor graphs. The effect depends on graph density, message-passing depth, aggregation strategy, and the location of the corrupted sensor. Physics-informed constraints, adaptive edge weighting, denoising modules, and sensor-confidence estimates can reduce the influence of unreliable measurements, although these techniques do not eliminate the risk under severe or coordinated corruption [56, 57].

11.4. Open-Set Fault Misclassification and Novelty-Rejection Failure

Closed-set GNN classifiers assign each input to one of the fault categories observed during training, which may cause previously unseen or out-of-distribution faults to be mapped to known classes with high confidence [57, 58]. In marine machinery, wind-energy systems, and industrial processes, novel degradation patterns, compound faults, or rare operating conditions may therefore produce incorrect diagnoses and inappropriate maintenance responses. Semi-supervised open-set GNNs, distance-based rejection, uncertainty thresholds, and novelty-detection mechanisms can

improve unknown-fault identification. However, threshold selection remains sensitive to class imbalance, calibration quality, and overlap between weak known faults and unknown conditions. A threshold that is too permissive may accept novel faults as known, whereas an overly restrictive threshold may reject difficult but valid known-class samples.

11.5. Oversmoothing, Oversquashing, and Dilution of Fault Signatures

Increasing message-passing depth can cause node representations to become progressively similar, a phenomenon commonly described as oversmoothing. This loss of representational distinction may obscure weak or incipient fault signatures that depend on subtle differences among neighboring components [59, 60]. The problem is particularly relevant to large graphs and prognostic settings in which early-stage defects, such as minor bearing damage or gradual insulation degradation, generate only small deviations from normal operating behavior. A related limitation is oversquashing, which occurs when information from a rapidly expanding number of distant nodes must be compressed into fixed-dimensional node representations [61]. In cyber-physical systems with long-range dependencies or bottlenecked graph structures, this compression may prevent weak but operationally important fault information from propagating effectively across multiple graph hops. Consequently, faults originating in one subsystem may exert insufficient influence on distant but functionally connected components, reducing the model's ability to identify cascading failures, distributed anomalies, and long-range fault-propagation patterns.

Potential mitigation strategies include residual and skip connections, jumping-knowledge mechanisms, attention normalization, graph rewiring, positional or structural encodings, hierarchical pooling, and multiscale architectures. Residual connections and jumping-knowledge mechanisms preserve information from earlier layers, while graph rewiring and positional encodings improve communication between structurally distant nodes. Hierarchical pooling and multiscale representations support information aggregation across subsystem and system levels. Graph-Transformer architectures may further alleviate oversquashing by enabling direct interactions between distant nodes, although their computational and memory requirements can restrict their suitability for large-scale or edge-constrained deployments. The effectiveness of these strategies depends on graph topology, message-passing depth, representation dimensionality, and the spatial distribution of fault-related features. Therefore, GNN-based fault-diagnostic systems must balance receptive-field expansion with the preservation of local distinctions, long-range information flow, and computational feasibility.

11.6. Adversarial Attacks and Data-Poisoning Vulnerabilities

Networked GNN-based diagnostic systems may be exposed to adversarial manipulation of node features, sensor measurements, graph edges, or training data. Structure poisoning, targeted feature perturbation, and backdoor attacks can produce false negatives that conceal faults or false positives that trigger unnecessary interventions [62, 63]. Such

risks are particularly relevant in power, transportation, and industrial-control systems because diagnostic outputs may influence automated control or maintenance decisions [64]. Proposed defenses include adversarial training, certified robustness, causal representation learning, anomaly filtering, uncertainty-based rejection, and validation of graph updates. However, practical defenses remain limited by computational cost, incomplete threat models, adaptive attackers, and the difficulty of distinguishing malicious manipulation from genuine changes in system behavior.

11.7. Concept Drift and Long-Term Model Obsolescence

Equipment aging, seasonal load variation, component replacement, maintenance activities, changing production regimes, and gradual wear can alter the relationship between input measurements and fault labels over time. A GNN trained on historical data may therefore experience reduced accuracy, increased false-alarm rates, and weaker calibration as operating conditions change. Static models are especially vulnerable when feature distributions and graph relationships evolve simultaneously. Continual learning, domain adaptation, drift detection, periodic recalibration, and federated model updating have been proposed to support long-term adaptation [10, 65]. These approaches must nevertheless control catastrophic forgetting, prevent erroneous updates from contaminating the model, and distinguish persistent concept drift from short-term disturbances.

These failure modes show that performance on static benchmarks does not by itself establish operational dependability. Error propagation, graph mismatch, noise amplification, open-set misclassification, oversmoothing, adversarial manipulation, and concept drift require complementary mitigation strategies, including adaptive graph learning, uncertainty estimation, robust aggregation, continual adaptation, open-set evaluation, and security testing. Addressing these risks requires coordination among graph learning, control engineering, cybersecurity, reliability analysis, and human oversight when GNN-based diagnostic models are incorporated into safety-critical cyber-physical systems.

12. Research Trends and Statistical Analysis

Research on GNN-based fault diagnosis has expanded rapidly in publication volume, application scope, and architectural diversity [3, 7, 22]. Since 2020, increasing attention has been directed toward spatiotemporal, attention-based, adaptive, causal, and hybrid models designed for interconnected and dynamically changing systems [6, 17]. However, quantitative synthesis remains constrained by differences in datasets, graph construction, preprocessing, fault definitions, operating conditions, and evaluation protocols. This section therefore examines publication trends, representative performance evidence, architecture-level trade-offs, and the limitations affecting cross-study statistical comparison.

12.1. Publication Trends and Bibliometric Insights

A bibliometric analysis of 1,495 Web of Science publications from 1998 to 2024 reported an annual growth rate of approximately 13.58% in AI-supported fault-diagnosis

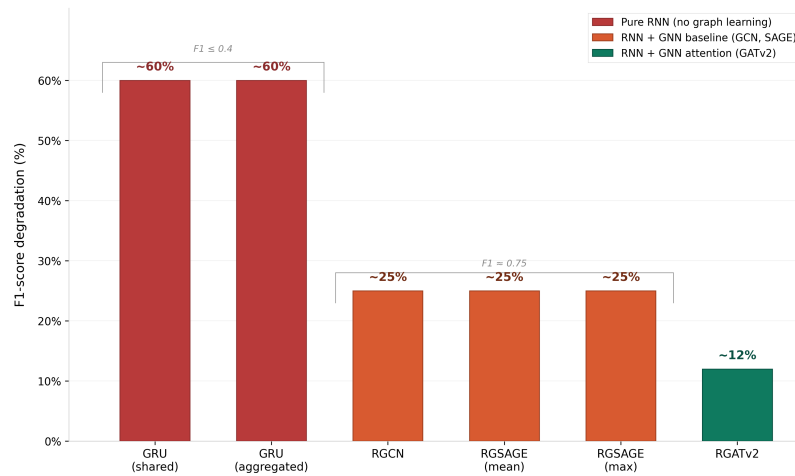


Figure 6: F1-score degradation under topology change for RNN and RNN+GNN pipeline architectures on the IEEE 123-node distribution network. Models were trained using an 11-PMU configuration and evaluated using 25 PMUs, most of which were unseen during training. The attention-based RGATv2 architecture exhibited the lowest reported degradation, at approximately 12% [11].

research, with China and the United States accounting for 35.34% and 18.27% of the analyzed output, respectively [3]. The acceleration observed after 2020 coincides with the expansion of Industrial Internet of Things infrastructures, predictive-maintenance applications, renewable-energy systems, and sensor-rich cyber-physical environments. Research themes have progressively shifted from conventional bearing and gearbox diagnosis toward distributed monitoring, dynamic graph modeling, knowledge-enhanced diagnosis, and system-level predictive maintenance.

12.2. Methodological Evolution and Performance Advances

Earlier studies primarily employed graph convolutional networks for static graph structures and vibration-based fault classification, whereas recent research increasingly emphasizes spatiotemporal, attention-based, adaptive, causal, and hybrid architectures [14, 22]. Representative evidence indicates that graph-based relational modeling can improve robustness under structural variation, although reported benefits remain dependent on the experimental setting, graph-construction strategy, and comparison baselines.

On the IEEE 123-node distribution network, a recurrent GRU model exhibited F1-score degradation approaching 60% when evaluated under a substantially altered PMU configuration. RNN+GNN pipelines using GCN and GraphSAGE reduced this degradation to approximately 25%, while the attention-based RGATv2 architecture limited it to approximately 12% [11]. Figure 6 illustrates this comparison and indicates the potential value of topology-aware attention under the evaluated conditions. These results should not, however, be interpreted as evidence of universal robustness because they are specific to the IEEE 123-node network, the selected PMU configurations, and the applied perturbation protocol.

Other reported advances include causal and knowledge-enhanced GNNs for reducing sensitivity to spurious relationships, few-shot and adaptive methods for diagnosis under limited labels, and multiscale architectures for weak fault signatures. For example, a few-shot graph-learning framework reported an accuracy of 94.32% for wind-turbine current-signal diagnosis under its specified experimental

conditions [30]. Such findings indicate progress in robustness, data efficiency, and adaptive diagnosis, but they should not be generalized across incompatible datasets, operating regimes, or evaluation protocols.

12.3. Comparative Analysis of Representative GNN-Based Methods

Table 5 compares representative GNN-based fault-diagnostic studies across architectures, application domains, datasets, graph-construction strategies, diagnostic objectives, reported results, computational information, strengths, and limitations. The reported values correspond to the original experimental settings and are not directly comparable because the studies differ in preprocessing, fault classes, graph definitions, operating conditions, train-test procedures, and evaluation metrics. Computational information is included when explicitly reported; otherwise, it is marked as not reported (NR). This distinction is important because high predictive performance does not necessarily imply suitability for real-time, large-scale, or resource-constrained deployment.

The comparison indicates that architecture selection involves trade-offs rather than a universally superior model. Conventional GCNs provide relatively efficient neighborhood aggregation and are suitable when graph structure is stable and well defined, but they remain sensitive to graph-construction errors and may dilute weak local distinctions as depth increases. Graph Attention Networks can assign different importance to neighboring nodes and may improve robustness when relationships vary in relevance; however, attention computation introduces additional parameters, memory demand, and inference cost.

Spatiotemporal GNNs combine structural and temporal modeling and are therefore appropriate for sensor streams, evolving degradation, and time-dependent fault propagation. Their computational requirements increase with graph size, sequence length, temporal-window size, and message-passing depth, which can restrict real-time deployment. Hypergraph GNNs represent higher-order relationships among groups of components and may capture interactions that pairwise graphs cannot express, but hy-

Table 5: Comparative analysis of representative GNN-based fault-diagnostic methods. Results correspond to the original experimental settings and should not be interpreted as directly comparable across heterogeneous datasets and protocols.

Study	Architecture	Domain	Dataset / system	Graph construction	Task	Metrics and reported result	Strengths	Limitations
Study [23]	GCN	Industrial process diagnosis	Study-specific measurement dataset	Prior engineering knowledge and measurement relationships	Classification	Reported improved diagnostic performance over conventional baselines	Integrates relational measurements with engineering knowledge	Sensitive to graph-prior quality; limited cross-domain validation
Study [24]	Graph-based multivariate model	Power-transformer diagnosis	Dissolved-gas-analysis data	Relationships among dissolved-gas variables	Classification	Reported stronger performance than selected conventional and machine-learning baselines	Models dependencies among multiple dissolved-gas measurements	Study-specific evaluation; limited scalability and deployment reporting
Study [26]	Physics-informed GNN	Power distribution systems	IEEE 123-node and 37-node feeders	Physical grid topology and similarity among labeled and unlabeled samples	Localization	Reported improved localization under limited labels, load variation, and topology changes; the two-stage approach improved accuracy by up to approximately 6% over supervised training	Supports sparse observations, limited labels, and structural variation through physical priors	Depends on grid-model accuracy and simulated operating assumptions
Study [27]	Hierarchical attention GCN with BiLSTM	Aero-engine prognostics	Multisensor degradation dataset	Sensor-dependency graph with hierarchical attention	RUL estimation	Reported lower prediction error than selected comparison models under the original protocol	Combines intersensor dependencies with temporal degradation modeling	Graph construction and performance may be sensitive to operating-condition changes
Study [11]	GRU+GCN, GraphSAGE, and RGATv2	Power distribution systems	IEEE 123-node network; 11-PMU training and 25-PMU testing	Power-network topology with message passing and relational attention	Diagnosis under topology change	GRU degradation approached 60%; GCN and GraphSAGE pipelines degraded by approximately 25%; RGATv2 degradation was approximately 12%	Provides explicit evaluation under substantial topology and sensor variation	Results are specific to one network, PMU configuration, and perturbation protocol
Few-shot graph framework [30]	Few-shot adaptive GNN	Wind-turbine diagnosis	Current-signal dataset	Graph relationships among signal-derived features	Classification with limited labels	Reported accuracy of 94.32% under the specified experimental setting	Addresses limited labeled data and supports adaptive diagnosis	Dataset- and protocol-specific result; latency and scalability not established
DACDFE-GNN [8]	Dynamic multiscale GNN	Photovoltaic-integrated power systems	Study-specific photovoltaic fault dataset	Adaptive edge weighting with physical and topological representations	Weak-fault diagnosis	Reported stronger weak-fault recognition than selected baselines under noise and low-sampling conditions	Captures weak fault signatures and changing graph relationships	Dynamic updating may increase complexity and sensitivity to transient noise
Multi-metric fusion hypergraph network [16]	Hypergraph GNN	Rotating machinery	Study-specific multisource machinery dataset	Hyperedges based on instance, distributional, and spatiotemporal relationships	Classification	Reported improved classification over selected graph and deep-learning baselines	Represents higher-order relationships beyond pairwise connections	Hyperedge construction and scalability remain dataset-dependent

peredge construction is often application-specific and can increase memory use and propagation complexity. Graph-Transformer and hybrid architectures improve long-range dependency modeling and reduce reliance on local message passing, although their scalability depends on attention sparsification, graph size, model depth, and available computational resources. Architecture comparisons should therefore jointly consider predictive accuracy, robustness, latency, memory consumption, graph-construction cost, interpretability, and suitability for the intended deployment environment [14, 22].

12.4. Performance and Computational Comparison

Performance-based comparison across GNN-based fault-diagnostic studies is complicated by differences in datasets, graph-construction strategies, diagnostic objectives, train-test partitions, hardware platforms, and evaluation metrics. Consequently, reported values should not be interpreted as a direct ranking of architectures. Nevertheless, the available evidence permits a structured comparison of predictive performance, computational efficiency, latency, scalability, and resource requirements across major GNN categories. Table 6 summarizes these characteristics using representative results reported in the reviewed literature. Values that were not provided by the original studies are marked as not reported (NR).

The comparison shows that predictive measures are reported more consistently than operational and computational measures. Accuracy, F1-score, localization accuracy, and regression error are commonly provided, whereas inference latency, parameter count, floating-point operations, memory consumption, training time, and energy use are frequently omitted. Scalability is therefore often demonstrated indirectly through evaluation on larger graph sizes, multiple systems, changing sensor configurations, topology variation, cross-domain transfer, or robustness under limited labels and noise.

Conventional GCNs generally offer the lowest computational burden but remain sensitive to graph construction and depth-related degradation. Attention-based and dynamic models provide stronger adaptability and robustness, although their memory and runtime requirements increase with graph density and attention complexity. Spatiotemporal models improve dynamic fault representation but introduce additional temporal-processing cost. Hypergraph and Graph-Transformer architectures provide richer higher-order and long-range dependency modeling, but their computational requirements may restrict deployment on edge devices. Physics-informed models improve structural consistency and label efficiency, whereas federated models reduce data-centralization requirements at the cost of communication overhead and slower convergence. These

Table 6: Performance and computational comparison of representative GNN architecture categories used for fault diagnostics. NR denotes information not reported in the underlying studies. Results are not directly comparable because they were obtained using different datasets, tasks, graph structures, and experimental protocols.

Architecture category	Reported predictive performance	Efficiency and latency	Computational cost	Scalability and robustness evidence	Principal trade-off
Conventional GCN	Representative studies report improved classification and localization over conventional machine-learning and deep-learning baselines; in topology-change experiments, GCN-based pipelines showed approximately 25% degradation compared with approximately 60% for recurrent-only baselines [11, 23]	Sparse neighborhood aggregation is generally efficient, but exact inference latency is frequently NR	Computational cost grows with the number of graph edges and message-passing layers; parameter count, FLOPs, and memory are often NR	Evaluated on industrial-process data and distribution networks with more than 100 nodes; performance remains sensitive to topology mismatch and graph-prior quality	Relatively simple and efficient, but vulnerable to oversmoothing, inaccurate graph construction, and limited adaptation to structural change
Graph attention networks	Attention-based variants report stronger robustness than recurrent and conventional graph baselines under topology and sensor changes; relational GAT variants showed approximately 12% performance degradation in one cross-topology evaluation [11]	Attention improves selective aggregation but increases inference and memory overhead; exact latency is commonly NR	Cost increases with edge count, number of attention heads, and neighborhood size; detailed FLOPs and memory reporting remain limited	Demonstrated robustness under changing sensor configurations and previously unseen measurement locations	Improved neighbor weighting and robustness are obtained at the cost of greater memory use and computational complexity
Spatiotemporal GNN	Reported improvements include gains of up to approximately 11 % points in F1-score over recurrent baselines under partial-observability conditions [11]	Reduced measured-node graph configurations have achieved approximately sixfold lower training time; exact inference latency remains NR	Cost depends jointly on graph size, temporal-window length, recurrent or convolutional temporal modules, and message-passing depth	Evaluated under partial observability, changing operating conditions, noise, and dynamic system behavior	Strong temporal and relational modeling, but increased training cost and weaker suitability for low-resource deployment
Physics-informed and dynamic GNN	Representative power-system studies report fault-detection accuracy up to 100%, exact localization around 94.73%, one-hop localization around 99.67%, and classification around 97.48% under the reported settings [26]	Physical priors can reduce the search space, but dynamic graph updating and adaptive attention increase run-time; exact latency is NR	Three-layer backbones and long training schedules have been reported; computational overhead increases with graph size and dynamic attention groups	Evaluated across 123-node and 34-node feeders, low-label settings, 30–40 dB noise, topology changes, and cross-system transfer	High diagnostic accuracy and physical consistency, but dependence on reliable system models and increased adaptation cost
Hypergraph GNN	Representative models report classification accuracy of approximately 0.9965 ± 0.0025 using only 5% labeled data [16]	High label efficiency is demonstrated, but inference latency and run-time efficiency are generally NR	Additional cost arises from hyperedge construction and higher-order message passing; FLOPs, parameters, and memory are often NR	Evaluated across multiple rotating-machinery datasets and under noisy and limited-label conditions; large-graph scalability remains insufficiently studied	Captures higher-order dependencies effectively, but hypergraph construction is application dependent and computationally demanding
Graph-Transformer hybrids	Representative studies report reduced prediction or classification error relative to selected graph and temporal baselines under study-specific protocols [7]	Long-range attention can improve representation quality but generally introduces higher latency than local message passing; precise latency is frequently NR	Self-attention may scale quadratically with the number of tokens or nodes unless sparse or hierarchical attention is used	Applied to long temporal sequences and multisensor degradation modeling, but evidence from very large graphs and real-time edge deployment remains limited	Strong long-range dependency modeling, but high memory consumption and limited suitability for resource-constrained systems
Federated and distributed GNN	Predictive performance is generally reported as close to centralized baselines under selected non-IID and multi-client settings [66]	Local inference can be efficient, but training latency depends strongly on communication rounds, synchronization, and client availability	Cost includes local computation, repeated parameter exchange, communication bandwidth, and aggregation overhead	Supports multi-site learning without centralizing raw data; scalability is affected by client heterogeneity, communication bottlenecks, and changing local graph structures	Improved privacy and collaborative learning are accompanied by communication cost and convergence challenges

observations indicate that future studies should report predictive accuracy jointly with latency, training time, model size, memory consumption, hardware configuration, and performance as graph size increases.

12.5. Challenges in Statistical Evaluation and Future Directions

Direct statistical comparison across studies remains difficult because experimental protocols vary in dataset size, fault composition, graph-construction strategy, preprocessing, class balance, train-test partitioning, topology perturbation, and metric selection [18, 22]. Accuracy, F1-score, and AUC are commonly reported, but these measures alone may not adequately characterize open-set recognition, rare-fault performance, uncertainty calibration, computational efficiency, or robustness under changing operating conditions.

Another limitation is the inconsistent reporting of computational characteristics. Many studies provide predictive

metrics without reporting parameter count, training duration, inference latency, memory use, communication overhead, or energy consumption. This omission prevents reliable assessment of whether a model is suitable for large graphs, edge devices, real-time monitoring, or distributed deployment. Computational claims should therefore be supported by measurements obtained under clearly specified hardware, batch size, graph size, and implementation conditions.

Future evaluations should report dataset characteristics, graph definitions, preprocessing procedures, partitioning strategies, model complexity, parameter count, training cost, inference latency, memory requirements, and robustness under noise, topology variation, unknown faults, and distribution shift. Where federated or distributed architectures are used, communication volume, synchronization frequency, client participation, and convergence behavior should also be documented. Standardized public repos-

itories and consistent reporting protocols would enable more reliable comparison among GNN architectures and support statistically meaningful assessment across application domains. The benchmark and reproducibility issues underlying these requirements are examined further in Section 8.

13. Emerging Research Directions

Emerging research on Graph Neural Networks (GNNs) for fault diagnosis increasingly focuses on deployment-oriented limitations involving privacy, physical consistency, multimodal heterogeneity, structural adaptation, and robustness. Rather than increasing architectural complexity alone, future work must determine whether these methods remain reliable under realistic operating constraints, limited labels, evolving topologies, and distributed data environments.

13.1. Federated and Privacy-Preserving GNNs

Federated GNNs enable collaborative model development across distributed industrial sites without centralizing raw sensor data and have been investigated for graph-structured industrial data, non-independent and identically distributed client datasets, heterogeneous local topologies, and communication-constrained settings [53, 66]. Applications include rotor-system diagnosis, anomaly detection in time-varying networks, and causal federated graph learning. Future research should move beyond demonstrating distributed-training feasibility and establish communication-efficient, privacy-verifiable, and topology-aware aggregation under heterogeneous client conditions. Comparative studies should report communication volume, client participation, local graph variation, privacy assumptions, aggregation strategy, and convergence behavior alongside diagnostic performance.

13.2. Physics-Informed and Knowledge-Guided Graph Learning

Physics-informed and knowledge-guided GNNs incorporate physical laws, engineering constraints, causal structure, or expert knowledge into graph construction and message passing [67]. Such integration may reduce reliance on large labeled datasets and discourage physically inconsistent predictions, and recent work has combined physical topology, electrical measurements, adaptive edges, and dual-field representations in power-system and industrial diagnostic applications. However, performance depends on the validity and completeness of the incorporated knowledge, and incorrect or oversimplified assumptions may bias graph construction and restrict generalization under unseen operating conditions. Future research should evaluate sensitivity to misspecified constraints, quantify the contribution of physical priors, and distinguish improvements arising from domain knowledge from those produced by increased model complexity.

13.3. Multimodal and Cross-Domain Fusion

Multimodal graph learning combines heterogeneous sources such as vibration, acoustic, thermal, visual, electrical, and operational data to construct broader representations of system condition [68]. This direction may

improve diagnosis when fault evidence is distributed across modalities and may support transfer across related systems. Key challenges include temporal misalignment, missing or corrupted modalities, unequal sampling rates, feature incompatibility, and increased computational demand. Future work should examine modality reliability, incomplete-input behavior, cross-domain transfer, and failure cases in which an unreliable modality degrades the fused prediction, while also distinguishing gains from complementary information from those caused by larger model capacity.

13.4. Dynamic and Self-Evolving Graphs

Dynamic and self-evolving graph methods update graph structure as operating conditions, components, and system configurations change [8]. These approaches are relevant to network reconfiguration, component replacement, renewable-energy integration, transient operation, and evolving degradation patterns. The principal challenge is distinguishing persistent structural change from temporary measurement noise, because frequent graph updates may increase computational cost, create unstable representations, or introduce spurious edges. Future studies should therefore investigate change-detection criteria, update frequency, stability guarantees, continual adaptation, and the computational feasibility of online graph revision in real-time systems.

13.5. Robust and Adversarially Resilient Diagnostics

GNN-based diagnostic systems remain vulnerable to sensor spoofing, feature perturbation, graph manipulation, training-data contamination, and distribution shift [17]. Proposed defenses include adversarial training, causal interventions, invariant representations, uncertainty estimation, and rejection mechanisms, but existing results are often specific to a dataset, attack model, perturbation magnitude, or topology-change protocol. Future evaluations should consider combined feature and structural attacks, adaptive adversaries, unknown fault conditions, and trade-offs between clean-condition performance and robustness. Robustness reporting should include calibration, degradation under perturbation, computational overhead, and failure-detection capability rather than predictive accuracy alone.

Collectively, these directions shift attention from isolated performance improvements toward reliable operation under distributed data, incomplete knowledge, heterogeneous modalities, evolving structures, and adversarial conditions. Progress will require reproducible experiments, realistic deployment settings, transparent computational reporting, and evaluation protocols that assess both predictive performance and operational resilience.

14. Design Guidelines for Practitioners

Deploying Graph Neural Network (GNN)-based fault-diagnostic systems in operational cyber-physical environments requires consideration of data quality, graph construction, model architecture, computational constraints, security, and safety requirements. Drawing on recent research and the limitations, failure modes, and ethical considerations discussed in this survey, this section presents

practical guidelines for designing reliable, interpretable, and deployable GNN-based diagnostic systems [69].

14.1. Graph Construction and Validation

Graph construction should reflect the physical and functional relationships of the monitored system. Practitioners should incorporate available information about physical topology, control logic, component interactions, and causal relationships rather than relying exclusively on statistical correlations [70]. Hybrid strategies that combine physics-based priors with data-driven edge learning may support adaptation when system relationships are uncertain or change over time. Graph structures should also be reassessed after equipment replacement, maintenance, sensor failure, or network reconfiguration. Online topology estimation, consistency checks, and digital-twin simulations may help identify graph mismatch and reduce diagnostic errors caused by outdated connectivity assumptions.

14.2. Data Quality, Preprocessing, and Noise Mitigation

Sensor-data quality should be assessed before model training and during deployment because noise, calibration drift, missing values, and synchronization errors can affect both node features and learned graph relationships. Preprocessing pipelines should include appropriate filtering, outlier detection, sensor-calibration checks, missing-data handling, and temporal alignment [70]. Adaptive edge weighting or sensor-reliability scores can reduce the influence of unreliable measurements during message passing, although these mechanisms should be validated under realistic sensor-failure conditions. Recording sensor-health indicators, preprocessing transformations, data provenance, and imputation procedures can also improve traceability and support investigation of incorrect diagnostic outputs.

14.3. Model Architecture and Robustness Design

Model architecture should balance representational capacity, robustness, interpretability, and computational cost. Residual connections, attention mechanisms, graph rewiring, and multiscale aggregation may reduce over-smoothing and preserve local fault information in deeper GNNs [69]. Physics-informed constraints and causal regularization can be incorporated when reliable domain knowledge is available, but inaccurate constraints may introduce additional bias. Uncertainty estimation through ensembles, probabilistic outputs, Bayesian approximations, or calibrated confidence scores can help identify predictions that require further review. Architecture selection should therefore be based not only on benchmark accuracy but also on calibration, robustness to graph perturbations, computational requirements, and performance under changing operating conditions.

14.4. Handling Open-Set and Evolving Fault Conditions

Operational systems may encounter fault categories and degradation patterns that were absent from the training data. Open-set recognition, novelty detection, and uncertainty-based rejection mechanisms can reduce the risk of assigning unknown faults to inappropriate known categories [69]. Rejection thresholds should be calibrated

using validation data that include weak known faults, rare events, and representative unknown conditions because poorly selected thresholds can either accept novel faults or reject difficult known samples. Continual learning, domain adaptation, periodic retraining, and drift detection may support adaptation to changes in equipment condition and operating regimes. Synthetic data and digital-twin simulations can supplement rare-fault examples, but simulated scenarios should be validated to ensure that they represent plausible fault behavior and propagation patterns.

14.5. Deployment Strategy and Resource Awareness

Deployment architecture should be selected according to response-time requirements, privacy constraints, network availability, graph size, and available computing resources. Edge deployment may be appropriate for time-sensitive local inference, while cloud-based or hybrid edge-cloud architectures can support model training, historical analysis, and aggregation across facilities [71]. Pruning, quantization, knowledge distillation, graph sampling, and workload partitioning can reduce memory and computation requirements, although their effect on diagnostic accuracy and calibration should be measured. Federated learning may support collaboration among sites without centralizing raw data, but it introduces communication, convergence, graph-heterogeneity, and security concerns. Deployment studies should therefore report inference latency, memory use, communication volume, model size, update frequency, and performance under network disruption.

14.6. Security, Privacy, and Adversarial Resilience

Security controls should address manipulation of sensor values, graph structures, training data, model updates, and communication channels [72]. Relevant measures include authenticated communication, access control, input validation, monitoring for abnormal graph changes, secure model updates, and audit logging. Adversarial training, graph-sanitization procedures, and uncertainty-based rejection may reduce sensitivity to selected attacks, but their effectiveness depends on the assumed threat model. Differential privacy, secure aggregation, and encrypted communication may be considered when operational or personal data are sensitive, although these protections can introduce computational cost or reduce model utility. Security evaluation should include feature perturbation, edge manipulation, poisoning, backdoor behavior, and compromised federated participants where these threats are relevant.

14.7. Human-in-the-Loop and Operational Integration

GNN-based diagnostics should support human decision-making rather than automatically replace operator judgment, particularly in safety-critical settings. Human-in-the-loop procedures can allow operators to review diagnostic outputs, provide corrective feedback, request additional evidence, and override automated recommendations when necessary [73]. Integration with supervisory control and data acquisition systems, digital twins, maintenance platforms, and existing alarm-management workflows should preserve established operational responsibilities and escalation procedures. Diagnostic interfaces should present

the predicted fault, affected components, uncertainty or confidence information, relevant sensor evidence, and the basis of any recommended action. Operator feedback should also be recorded so that recurring model errors and interface limitations can be identified.

14.8. Validation, Benchmarking, and Continuous Monitoring

Validation should cover the operating conditions and failure scenarios expected during deployment. Models should be evaluated under sensor noise, missing measurements, graph changes, class imbalance, unknown faults, and distribution shifts using standardized datasets and protocols where available [69, 73]. Cross-condition and cross-system testing can provide additional evidence of generalization, while calibration and selective-prediction metrics can indicate whether confidence estimates remain reliable. After deployment, continuous monitoring should track diagnostic accuracy, false-alarm rates, calibration, latency, resource use, graph changes, and data drift. Model updates should follow documented approval, validation, rollback, and audit procedures so that changes do not introduce unobserved performance degradation.

These guidelines emphasize that predictive performance alone is insufficient for operational deployment. Graph quality, data integrity, uncertainty, computational efficiency, security, human oversight, and continuous validation should be considered jointly when developing GNN-based fault-diagnostic systems. Applying these practices can support the transition from experimental models to operational tools while maintaining traceability, reliability, and alignment with system-specific safety requirements.

15. Ethical and Safety Considerations in GNN-Based Fault Diagnostics

As Graph Neural Network (GNN)-based fault-diagnostic systems move toward deployment in safety-critical cyber-physical environments, including smart grids, industrial automation, transportation, healthcare monitoring, and energy infrastructure, their design must account for safety, transparency, accountability, privacy, fairness, robustness, and harm prevention. These systems may influence operational continuity, human well-being, environmental protection, and access to essential services. Recent work on trustworthy GNNs identifies privacy preservation, adversarial robustness, fairness, and explainability as central requirements for responsible graph-learning systems [74]. This section reviews the ethical and safety concerns that arise when GNN-based diagnostics are integrated into operational decision-making.

15.1. Safety-Critical Decision Making and Risk of Harm

GNN-based diagnostic systems may support decisions such as fault isolation in power grids, shutdown recommendations in manufacturing, rerouting in transportation, and alerts in biomedical monitoring. Incorrect outputs caused by cascading misdiagnosis, topology drift, noisy measurements, model uncertainty, or adversarial inputs may lead to equipment damage, service interruption, delayed intervention, or inappropriate control actions. In healthcare settings, false negatives may delay necessary treatment, while false positives may lead to unnecessary

follow-up procedures. Uncertainty estimation can help operators identify predictions that require further review, but uncertainty scores must be calibrated and interpreted within the operational context. Safety-by-design practices, including conservative thresholds, redundant validation, fail-safe behavior, human-in-the-loop escalation, and controlled fallback procedures, can reduce the likelihood that an uncertain prediction directly triggers a harmful action [75].

15.2. Transparency, Explainability, and Operator Trust

The internal reasoning of deep GNN architectures is often difficult to inspect, which can limit operator understanding, regulatory review, and post-incident analysis. Engineers and operators may need to determine which nodes, edges, sensor values, or graph relationships contributed to a diagnostic outcome before acting on it. Explainable GNN methods include instance-level techniques, such as gradient attribution, perturbation masks, and relevance propagation, as well as model-level approaches based on prototypes, recurring subgraphs, or symbolic rules [76]. Domain-oriented explanations may also compare learned relationships with physical topology, control logic, or known fault-propagation paths. However, a visually coherent explanation is not necessarily causal or physically valid. Explanations should therefore be assessed for fidelity, stability, domain consistency, and usefulness to the intended operator rather than treated as sufficient evidence of model correctness.

15.3. Data Privacy, Ownership, and Regulatory Compliance

GNN-based diagnostics may process proprietary operational data, infrastructure records, or personally identifiable information collected from distributed sensors and monitoring systems. Cross-facility data sharing can expose industrial processes, maintenance practices, or intellectual property, while healthcare and transportation applications may involve personal data subject to legal and sector-specific protections. Federated GNN training, differential privacy, secure aggregation, and secure multiparty computation can reduce the need to centralize raw data [77]. These methods nevertheless involve trade-offs among privacy protection, communication cost, computational complexity, and diagnostic performance. Privacy governance should specify data ownership, permitted uses, retention periods, access controls, and procedures for handling model updates or inferred information that may reveal sensitive operational patterns.

15.4. Bias, Fairness, and Unequal Risk Distribution

Bias may arise when training data underrepresent particular operating conditions, equipment types, geographic regions, sensor configurations, or user populations. A model trained primarily on modern, well-instrumented systems may perform less reliably on aging infrastructure, lower-cost equipment, or facilities with sparse sensing. In healthcare and public infrastructure, uneven performance may create unequal exposure to false alarms, delayed detection, or service disruption. Fairness-aware GNN methods, bias auditing, stratified evaluation, and representative dataset design can help identify performance differences

across relevant groups or operating contexts [78]. Fairness assessment should be tailored to the application because equal aggregate accuracy does not necessarily imply equal safety risk, and the relevant comparison groups may involve equipment classes, geographic regions, facilities, or demographic populations.

15.5. Robustness Against Adversarial Manipulation and Security Threats

GNN-based diagnostic systems connected to operational networks may be exposed to data poisoning, evasion attacks, graph-structure manipulation, compromised sensors, or backdoor insertion. Such attacks can conceal faults, create false alarms, alter fault localization, or influence automated responses. Proposed defenses include adversarial training, graph denoising, certified robustness, anomaly filtering, causal interventions, and validation of graph updates [79]. However, robustness results are often dependent on a specific attack model, dataset, and perturbation budget. Security assessment should therefore consider feature manipulation, edge insertion or deletion, poisoned training data, compromised federated participants, and attacks that combine several channels. Diagnostic systems should also include authenticated communication, access control, audit logging, and procedures for isolating suspicious inputs or model updates.

15.6. Accountability, Liability, and Governance Frameworks

When a GNN-based diagnostic recommendation contributes to an operational failure or harmful decision, responsibility may be distributed among model developers, equipment vendors, system integrators, operators, data providers, and organizational decision-makers. This makes accountability difficult, particularly when model behavior cannot be reconstructed after an event. Human-in-the-loop procedures, version-controlled models, decision logs, data-provenance records, and documented override mechanisms can support incident investigation and traceability. Governance frameworks should define who approves model deployment, who monitors performance, who may override automated recommendations, and how errors are reported and corrected [75]. Liability and certification requirements will vary across sectors and jurisdictions, so technical controls should be aligned with the applicable safety standards, regulatory obligations, and organizational responsibilities.

15.7. Ethical Design Principles and Pathways to Trustworthy Deployment

Responsible deployment requires ethical and safety considerations to be incorporated throughout the system life-cycle rather than added after model development. Relevant practices include safety-by-design architectures, calibrated uncertainty thresholds, physically grounded explanations, privacy-by-default data handling, fairness audits, adversarial testing, human oversight, and continuous validation under changing operating conditions [74]. Interdisciplinary review involving engineers, domain specialists, safety professionals, legal experts, and affected stakeholders can help identify risks that are not captured by predictive metrics alone. GNN-based diagnostic systems should therefore be evaluated not only for accuracy but also for calibration,

failure containment, traceability, privacy, robustness, and consistency with application-specific safety requirements.

16. Conclusion

Graph Neural Networks have expanded the capabilities of fault-diagnostic systems by supporting relational and system-level modeling of interconnected components in complex cyber-physical environments. This survey synthesized developments in GNN-based fault diagnostics through a unified taxonomy, a review of benchmark datasets and evaluation practices, and an analysis of deployment architectures involving trade-offs among latency, scalability, privacy, and computational resources. The distinction between methodological limitations and operational failure modes—including cascading misdiagnosis, topology drift, noise amplification, open-set failures, adversarial vulnerabilities, and concept drift—highlights the importance of robustness, uncertainty awareness, and failure containment in real-world deployments. Emerging directions, including physics-informed modeling, federated learning, multimodal fusion, and adaptive graph evolution, indicate a continuing shift toward diagnostic frameworks that account for deployment constraints, data heterogeneity, and changing system conditions. Ethical considerations and practitioner-oriented design guidelines further show that predictive performance alone is insufficient for operational use. Future work should prioritize standardized benchmarks, transparent reporting, uncertainty-aware inference, cross-domain validation, computational efficiency, and interdisciplinary evaluation to support the development of reliable and safety-conscious GNN-based fault-diagnostic systems.

Funding

This research received no external funding.

Author Contributions

Conceptualization: Vaibhavi, Ramy, Anand;
Methodology: Vaibhavi, Ola;
Writing – Original Draft: Vaibhavi, Ola, Anand;
Writing – Review & Editing: Vaibhavi, Ola, Anand;
Visualization - Vaibhavi, Ramy;

Conflicts of Interest

The authors declare no conflict of interest.

References

- [1] Siemens AG, "The true cost of downtime 2024: How much do leading manufacturers lose through inefficient maintenance?", Tech. rep., Siemens Digital Industries, 2024.
- [2] ABB, "Industrial downtime costs up to \$500,000 per hour and can happen every week", <https://new.abb.com/news/detail/130589/industrial-downtime-costs-up-to-500000-per-hour-and-can-happen-every-week>, 2025, accessed: 2026-06-21.

- [3] Q. Li, Z. Wu, "Knowledge graph-enhanced fault diagnosis: A bibliometric review of ai applications in sensor management (1998–2024)", *Discover Artificial Intelligence*, vol. 5, p. 417, 2025, doi:[10.1007/s44163-025-00688-w](https://doi.org/10.1007/s44163-025-00688-w).
- [4] DataIntelto, "Graph neural networks market research report 2023", <https://dataintelto.com/report/graph-neural-networks-market>, 2025, reports global GNN market size of USD 1.45 billion in 2024 and projected USD 15.47 billion by 2033 at 30.2% CAGR; accessed 2026-06-21.
- [5] Precedence Research, "Neural network market size, share, and trends 2025 to 2034", <https://www.precedenceresearch.com/neural-network-market>, 2025, accessed: 2026-06-21.
- [6] Y. Liu, B. Shen, D. S.-H. Wong, M. Jia, Y. Yao, "Explainable neural network meets graph neural network: Recent advances in process fault detection and diagnosis", *Computers & Chemical Engineering*, vol. 206, p. 109528, 2026, doi:[10.1016/j.compchemeng.2025.109528](https://doi.org/10.1016/j.compchemeng.2025.109528).
- [7] S. Yang, R. Liu, "A review of graph neural networks for rolling bearing fault diagnosis", *Measurement Science and Technology*, 2026, doi:[10.1088/1361-6501/ae2e29](https://doi.org/10.1088/1361-6501/ae2e29).
- [8] J. Liu, Y. Huang, K. Chen, G. Liu, J. Yan, S. Chen, Y. Xie, Y. Yu, T. Huang, "Graph neural networks for fault diagnosis in photovoltaic-integrated distribution networks with weak features", *Sensors*, vol. 25, no. 18, p. 5691, 2025, doi:[10.3390/s25185691](https://doi.org/10.3390/s25185691).
- [9] W. Ouyang, Y. Jin, "Fault diagnosis of process systems based on graph neural network", "Proceedings of the 1st International Conference on Data Mining, E-Learning, and Information Systems (DMEIS 2024)", pp. 37–45, SCITEPRESS, 2024, doi:[10.5220/0012876300004536](https://doi.org/10.5220/0012876300004536).
- [10] M. Aurangzeb, Y. Wang, S. Iqbal, M. Shafiullah, S. A. Mohammed, Z. M. S. Elbarbary, A. Rehman, "Robust fault detection and uncertainty quantification in smart grids using graph neural networks", *Energy Reports*, vol. 15, p. 108920, 2026, doi:[10.1016/j.egyr.2025.12.057](https://doi.org/10.1016/j.egyr.2025.12.057).
- [11] B. Karabulut, C. Manna, C. Develder, "Generalization of graph neural network models for distribution grid fault detection", "2025 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)", pp. 1–7, IEEE, 2025, doi:[10.1109/SmartGridComm65349.2025.11204594](https://doi.org/10.1109/SmartGridComm65349.2025.11204594).
- [12] X. Wang, H. Wang, M. Peng, "Interpretability study of a typical fault diagnosis model for nuclear power plant primary circuit based on a graph neural network", *Reliability Engineering & System Safety*, vol. 261, p. 111151, 2025, doi:[10.1016/j.res.2025.111151](https://doi.org/10.1016/j.res.2025.111151).
- [13] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, R. Chou, J. Glanville, J. Grimshaw, A. Hróbjartsson, M. M. Lalu, T. Li, E. W. Loder, E. Mayo-Wilson, S. McDonald, L. A. McGuinness, L. A. Stewart, J. Thomas, A. C. Tricco, V. A. Welch, P. Whiting, D. Moher, "The prisma 2020 statement: An updated guideline for reporting systematic reviews", *BMJ*, vol. 372, p. n71, 2021, doi:[10.1136/bmj.n71](https://doi.org/10.1136/bmj.n71).
- [14] Z. Chen, J. Xu, C. Alippi, S. X. Ding, Y. A. W. Shardt, T. Peng, C. Yang, "Graph neural network-based fault diagnosis: A review", *arXiv preprint arXiv:2111.08185*, 2021, doi:[10.48550/arXiv.2111.08185](https://doi.org/10.48550/arXiv.2111.08185).
- [15] M. Jia, Y. Yao, Y. Liu, "Review on graph neural networks for process soft sensor development, fault diagnosis, and process monitoring", *Industrial & Engineering Chemistry Research*, vol. 64, no. 17, pp. 8543–8564, 2025, doi:[10.1021/acs.iecr.5c00283](https://doi.org/10.1021/acs.iecr.5c00283).
- [16] J. Zhu, J. Hu, B. Sheng, "Multi-metric fusion hypergraph neural network for rotating machinery fault diagnosis", *Actuators*, vol. 14, no. 5, p. 242, 2025, doi:[10.3390/act14050242](https://doi.org/10.3390/act14050242).
- [17] R. Liu, Q. Zhang, D. Lin, W. Zhang, S. X. Ding, "Causal intervention graph neural network for fault diagnosis of complex industrial processes", *Reliability Engineering & System Safety*, vol. 251, p. 110328, 2024, doi:[10.1016/j.res.2024.110328](https://doi.org/10.1016/j.res.2024.110328).
- [18] W. Xiao, Y. Wan, Z. Wang, C. Chen, "Spatio-temporal graph neural networks for fault diagnosis modeling of industrial robot", "Proceedings of the 31st International Conference on Engineering, Technology, and Innovation (ICE/ITMC)", pp. 1–9, IEEE, 2025, doi:[10.1109/ICE/ITMC65658.2025.11106644](https://doi.org/10.1109/ICE/ITMC65658.2025.11106644).
- [19] B. Wang, M. Wang, Y. Xu, L. Wang, S. Chen, X. Chen, "A diagnosis method based on graph neural networks embedded with multirelationships of intrinsic mode functions for multiple mechanical faults", *Defence Technology*, vol. 50, pp. 364–373, 2025, doi:[10.1016/j.dt.2025.04.014](https://doi.org/10.1016/j.dt.2025.04.014).
- [20] C. Li, L. Mo, C. K. Kwok, X. Li, Z. Chen, M. Wu, R. Yan, "Noise-robust multi-view graph neural network for fault diagnosis of rotating machinery", *Mechanical Systems and Signal Processing*, vol. 224, p. 112025, 2025, doi:[10.1016/j.ymssp.2024.112025](https://doi.org/10.1016/j.ymssp.2024.112025).
- [21] G. Jiang, K. Shen, X. Liu, X. Cheng, P. Xie, "Hierarchical spatio-temporal graph network for fault diagnosis of industrial processes", *IEEE Internet of Things Journal*, vol. 12, no. 3, pp. 3043–3054, 2025, doi:[10.1109/JIOT.2024.3476287](https://doi.org/10.1109/JIOT.2024.3476287).
- [22] Y. Wang, M. Wu, X. Li, L. Xie, Z. Chen, "A survey on graph neural networks for remaining useful life prediction: Methodologies, evaluation and future trends", *arXiv preprint arXiv:2409.19629*, 2024, doi:[10.48550/arXiv.2409.19629](https://doi.org/10.48550/arXiv.2409.19629).
- [23] Z. Chen, J. Xu, T. Peng, C. Yang, "Graph convolutional network-based method for fault diagnosis using a hybrid of measurement and prior knowledge", *IEEE Transactions on Cybernetics*, vol. 52, no. 9, pp. 9157–9169, 2022, doi:[10.1109/TCYB.2021.3059002](https://doi.org/10.1109/TCYB.2021.3059002).
- [24] W. Liao, D. Yang, Y. Wang, X. Ren, "Fault diagnosis of power transformers using graph convolutional network", *CSEE Journal of Power and Energy Systems*, vol. 7, no. 2, pp. 241–249, 2021, doi:[10.17775/CSEEJPES.2020.04120](https://doi.org/10.17775/CSEEJPES.2020.04120).
- [25] Q.-H. Ngo, B. L. H. Nguyen, J. Zhang, K. Schoder, H. Ginn, T. Vu, "Deep graph neural network for fault detection and identification in distribution systems", *Electric Power Systems Research*, vol. 247, p. 111721, 2025, doi:[10.1016/j.eprsr.2025.111721](https://doi.org/10.1016/j.eprsr.2025.111721).
- [26] W. Li, D. Deka, "Physics-informed graph neural networks for robust fault location in power grids", "ICML 2021 Workshop on Tackling Climate Change with Machine Learning", 2021.
- [27] T. Li, Z. Zhao, C. Sun, R. Yan, X. Chen, "Hierarchical attention graph convolutional network to fuse multi-sensor signals for remaining useful life prediction", *Reliability Engineering & System Safety*, vol. 215, p. 107878, 2021, doi:[10.1016/j.res.2021.107878](https://doi.org/10.1016/j.res.2021.107878).
- [28] Q. Wang, B. Han, "Temporal transaction network anomaly detection for industrial internet of things with federated graph neural networks", *Computers & Industrial Engineering*, vol. 205, p. 111122, 2025, doi:[10.1016/j.cie.2025.111122](https://doi.org/10.1016/j.cie.2025.111122).
- [29] Z. He, Y. Zeng, H. Shao, B. Yang, "Noise-robust rotating machinery fault diagnosis method based on enhanced graph isomorphism network with multi-sensor signals", *Measurement Science and Technology*, 2025, doi:[10.1088/1361-6501/adf45d](https://doi.org/10.1088/1361-6501/adf45d), metadata retained from submitted file; DOI-title match should be checked manually before submission.
- [30] Y. Wang, J. Zhao, D. Tang, W. Zhao, S. Huang, "Intelligent fault prediction and diagnosis for wind-powered heating systems using graph neural networks", *Scientific Reports*, vol. 15, p. 39068, 2025, doi:[10.1038/s41598-025-25884-7](https://doi.org/10.1038/s41598-025-25884-7).
- [31] M. Vaida, Z. Huang, "Multimodal graph neural networks in healthcare: A review of fusion strategies across biomedical domains", *Frontiers in Artificial Intelligence*, vol. 8, p. 1716706, 2026, doi:[10.3389/frai.2025.1716706](https://doi.org/10.3389/frai.2025.1716706).
- [32] C. Lou, M. A. Atoui, X. Zhang, D. Xu, H. Zhong, "Semi-supervised graph neural networks for fault diagnosis in marine machinery", *Engineering Applications of Artificial Intelligence*, vol. 177, no. 2, p. 114951, 2026, doi:[10.1016/j.engappai.2026.114951](https://doi.org/10.1016/j.engappai.2026.114951).
- [33] R. Bourgerie, T. Zanolida, "Fault detection in telecom networks using bi-level federated graph neural networks", "2023 IEEE International Conference on Data Mining Workshops (ICDMW)", pp. 1608–1617, IEEE, 2023, doi:[10.1109/ICDMW60847.2023.10449399](https://doi.org/10.1109/ICDMW60847.2023.10449399).
- [34] V. P. Dwivedi, C. K. Joshi, A. T. Luu, T. Laurent, Y. Bengio, X. Bresson, "Benchmarking graph neural networks", *Journal of Machine Learning Research*, vol. 24, no. 43, pp. 1–48, 2023.

- [35] A. Varbella, K. Amara, B. Gjorgiev, M. El-Assady, G. Sansavini, "Powergraph: A power grid benchmark dataset for graph neural networks", *Advances in Neural Information Processing Systems*, vol. 37, pp. 110784–110804, 2024, doi:10.52202/079017-3517.
- [36] Case Western Reserve University Bearing Data Center, "Bearing data center", <https://engineering.case.edu/bearingdatacenter>, accessed: 2026-06-13.
- [37] C. Lessmeier, J. K. Kimotho, D. Zimmer, W. Sextro, "Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification", *Proceedings of the European Conference of the Prognostics and Health Management Society*, vol. 3, 2016, doi:10.36001/phme.2016.v3i1.1577.
- [38] J. Lee, H. Qiu, G. Yu, J. Lin, R. T. Services, "Ims, university of cincinnati bearing data set", NASA Prognostics Data Repository, NASA Ames Research Center, 2007.
- [39] H. Qiu, J. Lee, J. Lin, G. Yu, "Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics", *Journal of Sound and Vibration*, vol. 289, no. 4–5, pp. 1066–1090, 2006, doi:10.1016/j.jsv.2005.03.007.
- [40] Y. Lei, T. Han, B. Wang, N. Li, T. Yan, J. Yang, "Xjtu-sy rolling element bearing accelerated life test datasets: A tutorial", *Journal of Mechanical Engineering*, vol. 55, no. 16, pp. 1–6, 2019, doi:10.3901/JME.2019.16.001.
- [41] B. Wang, Y. Lei, N. Li, N. Li, "A hybrid prognostics approach for estimating remaining useful life of rolling element bearings", *IEEE Transactions on Reliability*, vol. 69, no. 1, pp. 401–412, 2020, doi:10.1109/TR.2018.2882682.
- [42] H. Purohit, R. Tanabe, K. Ichige, T. Endo, Y. Nikaido, K. Suefusa, Y. Kawaguchi, "Mimii dataset: Sound dataset for malfunctioning industrial machine investigation and inspection", *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, pp. 209–213, 2019, doi:10.33682/m76f-d618.
- [43] R. Tanabe, H. Purohit, K. Dohi, T. Endo, Y. Nikaido, T. Nakamura, Y. Kawaguchi, "Mimii due: Sound dataset for malfunctioning industrial machine investigation and inspection with domain shifts due to changes in operational and environmental conditions", *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 21–25, 2021, doi:10.1109/WASPAA52581.2021.9632774, also available as arXiv:2105.02702.
- [44] A. Saxena, K. Goebel, "Turbofan engine degradation simulation data set", NASA Prognostics Data Repository, NASA Ames Research Center, 2008.
- [45] A. Saxena, K. Goebel, D. Simon, N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation", *2008 International Conference on Prognostics and Health Management*, pp. 1–9, 2008, doi:10.1109/PHM.2008.4711414.
- [46] J. J. Downs, E. F. Vogel, "A plant-wide industrial process control problem", *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993, doi:10.1016/0098-1354(93)80018-I.
- [47] S. Ghamizi, A. Bojchevski, A. Ma, J. Cao, "Safepowergraph: Safety-aware evaluation of graph neural networks for transmission power grids", *arXiv preprint arXiv:2407.12421*, 2024, doi:10.48550/arXiv.2407.12421.
- [48] Z. Zhu, F. Li, Z. Mo, Q. Hu, G. Li, Z. Liu, X. Liang, J. Cheng, " a^2q : Aggregation-aware quantization for graph neural networks", *International Conference on Learning Representations*, 2023.
- [49] A. Zhou, J. Yang, T. Qiao, Y. Qi, Z. Yang, W. Zhao, C. Hu, "Graph neural networks automated design and deployment on device-edge co-inference systems", *Proceedings of the 61st ACM/IEEE Design Automation Conference*, DAC '24, Association for Computing Machinery, New York, NY, USA, 2024, doi:10.1145/3649329.3655938.
- [50] J. Qin, R. Yang, N. Yu, "Physics-informed graph neural networks for collaborative dynamic reconfiguration and voltage regulation in unbalanced distribution systems", *IEEE Transactions on Industry Applications*, vol. 61, no. 2, pp. 2538–2548, 2025, doi:10.1109/TIA.2025.3529799.
- [51] A. Isah, H. Shin, I. Aliyu, R. M. Sulaiman, J. Kim, "Graph neural network for digital twin network: A conceptual framework", *2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, pp. 1–5, IEEE, 2024, doi:10.1109/ICAIIIC60209.2024.10463455.
- [52] S. Sheka, A. Saraswathi, "Improving mechanical fault diagnosis using graph neural networks with dynamic and multiscale features", *Engineering, Technology & Applied Science Research*, vol. 15, no. 4, pp. 25382–25387, 2025, doi:10.48084/etasr.11612.
- [53] Y. Wang, "Fedgnn-sfd: A lightweight federated graph neural network for multi-sensor bearing fault diagnosis in industrial iot systems", *Preprints.org*, 2026, doi:10.20944/preprints202603.2130.v1, preprint.
- [54] B. Chen, G. Lu, Y. Zhu, C. Liu, B. Qin, J. Li, "Gcfr: Graph contrastive fault representation for diagnosis in power communication networks", *Scientific Reports*, vol. 15, p. 39838, 2025, doi:10.1038/s41598-025-23457-2.
- [55] F. Wang, Y. Liu, K. Liu, Y. Wang, S. Medya, P. S. Yu, "Uncertainty in graph neural networks: A survey", *Transactions on Machine Learning Research*, 2025, doi:10.48550/arXiv.2403.07185.
- [56] Q. Zhou, L. Xue, J. He, S. Jia, Y. Li, "A rotating machinery fault diagnosis method based on dynamic graph convolution network and hard threshold denoising", *Sensors*, vol. 24, no. 15, p. 4887, 2024, doi:10.3390/s24154887.
- [57] S. Zhang, S. Shan, Z. Hu, Y. Shen, C. Li, K. Zhang, H. Wei, "Out-of-distribution fault detection in multi-sensor systems using spatio-temporal dynamic graph neural networks", *Mechanical Systems and Signal Processing*, vol. 241, p. 113524, 2025, doi:10.1016/j.ymssp.2025.113524.
- [58] J. Xu, Y. Wang, R. Xu, H. Wang, X. Zhou, "Research on open-set recognition methods for rolling bearing fault diagnosis", *Sensors*, vol. 25, no. 10, p. 3019, 2025, doi:10.3390/s25103019.
- [59] Y. Jin, X. Zhu, "Oversmoothing alleviation in graph neural networks: A survey and unified view", *Knowledge and Information Systems*, vol. 67, pp. 11259–11285, 2025, doi:10.1007/s10115-025-02548-6.
- [60] D. Kelesis, D. Fotakis, G. Paliouras, "Analyzing the effect of residual connections to oversmoothing in graph neural networks", *Machine Learning*, vol. 114, p. 184, 2025, doi:10.1007/s10994-025-06822-0.
- [61] U. Alon, E. Yahav, "On the bottleneck of graph neural networks and its practical implications", *International Conference on Learning Representations*, 2021.
- [62] A. Vassilev, A. Oprea, A. Fordyce, H. Anderson, X. Davies, M. Hamin, "Adversarial machine learning: A taxonomy and terminology of attacks and mitigations", *Tech. Rep. NIST AI 100-2e2025*, National Institute of Standards and Technology, 2025, doi:10.6028/NIST.AI.100-2e2025.
- [63] L. Gosch, M. Sabanayagam, D. Ghoshdastidar, S. Günnemann, "Provable robustness of (graph) neural networks against data poisoning and backdoor attacks", *Transactions on Machine Learning Research*, 2025.
- [64] V. Toğan, F. Mostofi, O. B. Tokdemir, "Evaluating the resilience of graph neural network architectures to adversarial and noisy data in high-stakes construction project management", *Journal of Construction Engineering and Management*, vol. 152, no. 4, p. 04026029, 2026, doi:10.1061/JCEMD4.COENG-17244.
- [65] Z. He, C. Shen, B. Chen, J. Shi, W. Huang, Z. Zhu, D. Wang, "A new feature boosting based continual learning method for bearing fault diagnosis with incremental fault types", *Advanced Engineering Informatics*, vol. 61, p. 102469, 2024, doi:10.1016/j.aei.2024.102469.
- [66] G. Mao, H. Li, L. Xue, Y. Li, Z. Cai, K. Noman, "Fedpm-sgn: A federated graph network for aviation equipment fault diagnosis by multi-sensor fusion in decentralized and heterogeneous setting", *Information Fusion*, vol. 117, p. 102876, 2025, doi:10.1016/j.inffus.2024.102876.

- [67] Z. Ma, C. Liu, F. He, G. Yang, "A physics-informed graph convolutional network for bearing fault diagnosis via dynamic constraint-guided feature learning", *Engineering Research Express*, vol. 8, p. 065524, 2026, doi:[10.1088/2631-8695/ae5166](https://doi.org/10.1088/2631-8695/ae5166).
- [68] S. Yu, X. Li, Y. Lei, B. Yang, N. Li, K. Feng, "Multimodal data-enabled large model for machine fault diagnosis towards intelligent operation and maintenance", *Journal of Industrial Information Integration*, vol. 50, p. 101061, 2026, doi:[10.1016/j.jii.2026.101061](https://doi.org/10.1016/j.jii.2026.101061).
- [69] T. Li, Z. Zhou, S. Li, C. Sun, R. Yan, X. Chen, "The emerging graph neural networks for intelligent fault diagnostics and prognostics: A guideline and a benchmark study", *Mechanical Systems and Signal Processing*, vol. 168, p. 108653, 2022, doi:[10.1016/j.ymssp.2021.108653](https://doi.org/10.1016/j.ymssp.2021.108653).
- [70] M. MansourLakouraj, M. Gautam, R. Hossain, H. Livani, M. Benidris, S. Commuri, "Event classification in active distribution grids using physics-informed graph neural network and pmu measurements", "2022 IEEE Industry Applications Society Annual Meeting (IAS)", pp. 1–6, IEEE, 2022, doi:[10.1109/IAS54023.2022.9939922](https://doi.org/10.1109/IAS54023.2022.9939922).
- [71] C. He, K. Balasubramanian, E. Ceyani, C. Yang, H. Xie, L. Sun, L. He, L. Yang, P. S. Yu, Y. Rong, P. Zhao, J. Huang, M. Annavaram, S. Avestimehr, "Fedgraphnn: A federated learning system and benchmark for graph neural networks", 2021, doi:[10.48550/arXiv.2104.07145](https://doi.org/10.48550/arXiv.2104.07145).
- [72] D. Z'ugner, O. Borchert, A. Akbarnejad, S. G'unnemann, "Adversarial attacks on graph neural networks: Perturbations and their patterns", *ACM Transactions on Knowledge Discovery from Data*, vol. 14, no. 5, pp. 57:1–57:31, 2020, doi:[10.1145/3394520](https://doi.org/10.1145/3394520).
- [73] D. Paolini, P. Dini, A. Elhanashi, S. Saponara, "Advanced fault detection and diagnosis exploiting machine learning and artificial intelligence for engineering applications", *Electronics*, vol. 15, no. 2, p. 476, 2026, doi:[10.3390/electronics15020476](https://doi.org/10.3390/electronics15020476).
- [74] H. Zhang, B. Wu, X. Yuan, S. Pan, H. Tong, J. Pei, "Trustworthy graph neural networks: Aspects, methods, and trends", *Proceedings of the IEEE*, vol. 112, no. 2, pp. 97–139, 2024, doi:[10.1109/JPROC.2024.3369017](https://doi.org/10.1109/JPROC.2024.3369017).
- [75] E. Tabassi, "Artificial intelligence risk management framework (airmf 1.0)", Tech. Rep. NIST AI 100-1, National Institute of Standards and Technology, 2023, doi:[10.6028/NIST.AI.100-1](https://doi.org/10.6028/NIST.AI.100-1).
- [76] H. Yuan, H. Yu, S. Gui, S. Ji, "Explainability in graph neural networks: A taxonomic survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, pp. 5782–5799, 2023, doi:[10.1109/TPAMI.2022.3204236](https://doi.org/10.1109/TPAMI.2022.3204236).
- [77] C. Wu, F. Wu, L. Lyu, T. Qi, Y. Huang, X. Xie, "A federated graph neural network framework for privacy-preserving personalization", *Nature Communications*, vol. 13, no. 1, p. 3091, 2022, doi:[10.1038/s41467-022-30714-9](https://doi.org/10.1038/s41467-022-30714-9).
- [78] A. Chen, R. A. Rossi, N. Park, P. Trivedi, Y. Wang, T. Yu, S. Kim, F. Deroncourt, N. K. Ahmed, "Fairness-aware graph neural networks: A survey", 2023, doi:[10.48550/arXiv.2307.03929](https://doi.org/10.48550/arXiv.2307.03929).
- [79] W. Jin, Y. Li, H. Xu, Y. Wang, S. Ji, C. Aggarwal, J. Tang, "Adversarial attacks and defenses on graphs: A review, a tool and empirical studies", 2020, doi:[10.48550/arXiv.2003.00653](https://doi.org/10.48550/arXiv.2003.00653).

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).



Vaibhavi Tiwari holds a MicroMasters credential in Data and Statistics from the Massachusetts Institute of Technology (MIT). She completed her bachelor's degree in Computer Applications in India and subsequently earned two master's degrees: a Master of Computer Applications from the National Institute of

Technology Karnataka (NITK), Surathkal, and a Master of Science in Computer Science from Montclair State University. With more than nine years of professional experience in healthcare technology, she has developed expertise in data security, artificial intelligence integration, cloud-based technologies, and large-scale data management, with a particular focus on secure, reliable, and intelligent solutions for healthcare and other critical sectors. Her research lies at the intersection of cybersecurity, healthcare informatics, artificial intelligence, cloud computing, and big data analytics, and she has published extensively in IEEE conferences, including recent works titled "Review of Ransomware Attacks and a Data Recovery Framework Using the Autopsy Digital Forensics Platform" and "Investigating Drone Data Recovery Beyond the Obvious Using Digital Forensics." These studies reflect her commitment to addressing emerging cybersecurity threats, strengthening digital-forensics practices, and advancing proactive approaches to threat identification, incident response, data recovery, and the protection of sensitive information in complex, data-driven environments. In addition to her research and professional work, Vaibhavi is a certified Globee Awards judge, recognizing her expertise in evaluating technology and business innovations, and she remains actively engaged in mentorship and community service through the Women in Big Data program and volunteer efforts with the International Science and Engineering Fair (ISEF), where she supports students and emerging professionals in science, technology, engineering, and data-related fields.



Ola Suaifan holds a Master's degree in Data Science from Montclair State University, a MicroMasters credential in Supply Chain Management from MIT, and a Bachelor's degree in Mathematics from the University of Jordan. She brings over eight years of experience in supply chain management, with current focus on integrating artificial intelligence into supply chain operations. Her research sits at the intersection of applied data science and data science systems development, with recent work exploring the integration of agentic AI in cybersecurity.



Ramy Othman is the Assistant Director of Technology and Infrastructure at the College of Science and Mathematics, Montclair State University, where he oversees GPU cluster management, Kubernetes orchestration, JupyterHub multi-user environments, and RAG pipeline development. He holds a Bachelor's degree in Electrical, Electronics and Communications Engineering from the Faculty of Engineering, Alexandria University, Egypt, and an MSc in Data Science from Montclair State University. In addition to his infrastructure role, he served as an Adjunct Professor in the School of Computing, delivering courses in Python Programming, Data Structures and Algorithms, Systems Programming, and Computer Networks. His research interests lie at the intersection of AI/ML infrastructure, cybersecurity, and applied machine learning, with a focus on scalable, GPU-

accelerated systems and multimodal AI pipelines. Othman earned First Prize in the Graduate Track of the RAISE-24 Informatics and Data Science Competition. He also built and manages the university's dedicated Network and AI research labs, through which he has mentored students.



Anand Gupta is a Senior Software Development Engineer at Amazon Web Services (AWS), an engineer, architect, and builder at heart. He is responsible for core billing systems that serve every AWS customer worldwide, having launched and expanded AWS and AWS

Marketplace billing across multiple regions and countries while driving invoicing modernization. He shaped multiple product vision to engineering design while at AWS. A relentless problem solver, Anand thrives on finding novel, inventive solutions that simplify complexity at scale. He believes great billing infrastructure should be invisible: accurate, reliable, and built to scale 100x, something customers never have to think about. Anand drives technical direction and influences engineering practices across an organization of 200+ engineers. He holds a Computer Science degree from IIIT Allahabad and has spent over 12 years at Amazon, building systems that scale. In his spare time, he enjoys playing badminton and tennis.