

UNIVERSITY OF OKLAHOMA GRADUATE COLLEGE

ADDRESSING ANOMALY DETECTION CHALLENGES IN AI-ENABLED MOBILE
AND SMART NETWORKS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the Degree of

DOCTOR OF PHILOSOPHY IN ELECTRICAL AND COMPUTER ENGINEERING

BY

ANEEQA IJAZ
Norman, Oklahoma
2025

ADDRESSING ANOMALY DETECTION CHALLENGES IN AI-ENABLED MOBILE
AND SMART NETWORKS

A DISSERTATION APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Ali Imran, Chair

Dr. James J. Sluss, Jr.

Dr. Samuel Cheng

Dr. Timothy G. Ford

© Copyright by ANEEQA IJAZ 2025

All Rights Reserved.

Acknowledgments

I am deeply grateful to Dr. Ali Imran, my supervisor, whose exceptional guidance and mentorship have been crucial in completing this dissertation. I will forever be indebted to Dr. Ali Imran for instilling in me the qualities of a diligent researcher and providing invaluable advice on excelling in both personal and professional aspects of life. Thank you, Dr. Imran, for being an exceptional mentor, kind leader, and for helping me at every stage of my academic and professional development.

I would also like to thank the committee members, Dr. James J. Sluss, Dr. Samuel Cheng, and Dr. Timothy G. Ford, for their valuable contributions. I express my sincere appreciation to Renee Wagenblatt, Denise Davis, Krista Pettersen, Josh Davis, Lee Lawson, Lauren Condry, and Monica Goodwin, for their exceptional support and assistance in handling all administrative tasks during my time as a PhD student at OU-Tulsa.

I am also immensely grateful to my family, who have been with me through all the highs and lows of this challenging journey. The contributions of my parents, in particular my father, and siblings cannot be overstated. Their prayers and constant encouragement have played a pivotal role in my success. I specifically extend my heartfelt thanks to my Husband, Momin Shahab, for his support, help, and belief in me. He was my consistent source of motivation. In particular, my son Rayaana M. Shahab, whose cheerful presence provided a persistent source of energy and strength to achieve the level of excellence in my research work. Their unending support, boundless love, and constant encouragement have served as my steadfast pillars of strength and inspiration throughout my academic life. I extend my sincere gratitude to my friends and colleagues at the AI4Networks Research Center, Dr. Farooq, Tabish, Dr. Manalastas, Waseem, Mehreen, Hifza, Umar, Dr. Masood, Dr. Ismail, Dr. Kasi. Their steadfast support and assistance have been invaluable during my academic journey.

Table of Contents

1	Introduction	1
1.1	Motivation	1
1.2	Problem Statements and Outline of the Proposed Solutions	2
1.3	Current and Planned Dissemination and Publications	4
1.4	Organization	7
2	Addressing the Challenges in AI-powered Diagnostics through Resource-Aware Hybrid Computing	8
2.1	Introduction	8
2.1.1	Motivation	8
2.1.2	Related Work	9
2.1.3	Contributions	10
2.1.4	Chapter Organization	11
2.2	Diagnostic Potential of Biomarkers Captured by Smart Technologies	12
2.2.1	AI-Driven Biomarker Feature Analysis for Disease Screening	14
2.3	AI-Enabled Biomarker Data Acquisition and Preliminary Screening	16
2.3.1	Cough Sensing Modalities	16
2.3.2	Open Access Datasets and Demographics	19
2.3.3	Signal Refinement for Acoustic Analysis	21
2.3.4	Feature Engineering for AI-Based Screening	22
2.4	Smartphone-Based Biomarker Collection and Health Screening	24
2.5	Challenges in Developing Smart, AI-Driven Screening Systems	28
2.6	SmartScreening: A Dual-Layered AI Framework for Smart and Reliable Diagnosis	30
2.7	Results and Discussion	32
2.8	Conclusion	40
3	EpiTrack: An AI-Driven Framework for Outbreak Forecasting by Leveraging Mobility Intelligence	42
3.1	Introduction	42
3.1.1	Motivation	42
3.1.2	Related Work	43

3.1.3	Contributions	46
3.1.4	Chapter Organization	47
3.2	Role of Wearable Devices in Biomarker Collection and Outbreak Monitoring	48
3.3	EpiTrack Framework	49
3.4	Simulation-Based Outbreak Prediction Architecture	54
3.4.1	Mobility Pattern Generation	54
3.4.2	Modeling Infection Propagation Dynamics	56
3.5	Implementation of Proactive AI Prediction Models	58
3.5.1	Feature-Based Learning for Exposure and Mobility Forecasting . .	59
3.5.2	Graph-Based Learning for Infection Dynamics	61
3.6	Results and Discussion	63
3.7	Conclusion	69
4	Proactive Reliability Modeling to Enable Resilient 6G Systems	71
4.1	Introduction	71
4.1.1	Motivation	71
4.1.2	Related Work	73
4.1.3	Contributions	77
4.1.4	Chapter Organization	78
4.2	Stochastic Modeling and Reliability Analysis	78
4.2.1	Model Development	81
4.2.2	Markov Analysis	83
4.2.3	Transition Matrix Properties	84
4.2.4	Transition Matrix Analysis	86
4.3	Numerical Results and Discussion	88
4.4	Utility of the Proposed Framework	94
4.5	Conclusion	96
5	An Intelligent Anomaly Detection Solution to Secure Multi-purpose Future Networks	98
5.1	Introduction	98
5.1.1	Motivation	98
5.1.2	Relevant Work	100
5.1.3	Contributions	104

5.1.4	Chapter Organization	104
5.2	Adversarial MDT Attacks	105
5.2.1	Vulnerability of Network Performance to MDT-based Attacks . .	106
5.2.2	Repercussions of MDT/CSI-based Attacks	107
5.3	MRIF Framework to Counter MDT Attacks	110
5.3.1	Anomaly Detection Module (ADM)	111
5.3.2	Malicious Reports Filtering Module (MRFM)	112
5.4	Results and Discussion	113
5.4.1	System model and analysis	114
5.5	Analysis and results for Anomaly Detection Module	115
5.5.1	Anomaly Detection Module with Autoencoder	115
5.5.2	Anomaly Detection Module with XGBoost	118
5.6	Analysis and results for Malicious Reports Filtering Module	119
5.7	Open Research Challenges	121
5.8	Conclusion	122
6	Conclusions and Future Work	124
6.1	Conclusion	124
6.2	Future Work	125
	References	128
	Funding Acknowledgements	154

List of Figures

2.1	Categorization of the AI techniques and performance evaluation parameters.	11
2.2	Cough types and the associated diseases.	13
2.3	AI-based cough detection/diagnosis process.	15
2.4	Customized cough detection and disease diagnosing applications.	25
2.5	Challenges faced during the incorporation of the AI techniques in health-care domain.	30
2.6	SmartScreening: an AI-based hybrid respiratory disease screening framework.	31
2.7	Comparison of model performance and feature importance	35
2.8	Impact of semi-supervised training under varying label availability.	36
2.9	Sensitivity and specificity comparison of deep learning models using Mel-spectrogram and spectrogram features.	40
3.1	Growth of wearable devices by region over the recent years.	48
3.2	Anomaly detection and outbreak prediction using biomarkers collected from smart devices.	49
3.3	EpiTrack : A mobile network-driven outbreak forecasting framework. . .	51
3.4	EpiTrack framework with detailed modules.	52
3.5	Open street map and SUMO zoomed in road map in Los Angeles.	55
3.6	Impact of varying inter-contact time (ICT) in the spread of the infection.	64
3.7	Feature importance for infection forecasting.	66
3.8	Comparative performance of GAT and Positional transformer models for infection risk prediction across key metrics.	68
3.9	Movement trends using LSTM	69
3.10	Impact of EpiTrack compliance on epidemic spread through mobility traces.	69
4.1	Misconfiguration probability of a single parameter as the number of configurable settings increases.	73
4.2	Suboptimal configuration of the Network Automation Function parameter.	79
4.3	Probabilistic tree illustrating NAF conflicts and resulting parameter state outcomes.	82
4.4	Probability of parameters to become suboptimal.	89
4.5	Impact of different parameters on network performance.	90

4.6	Steady state distribution of the network.	91
4.7	Steady state distributions versus misconfiguration probability.	92
4.8	Mean first passage time from state ‘S1’ to other states.	93
4.9	Parameter configuration scheduling prediction framework to avoid manual/network automation conflicts.	95
5.1	Classification of security threats in smart systems.	101
5.2	Adversarial MDT attack impact on the system’s SINR and network performance.	108
5.3	Malicious MDT Reports Identification Framework	111
5.4	Illustration of false MDT injection attacks and MRIF-based mitigation through cross-validation using FDT feedback.	113
5.5	System model with multi-cite cells where some sites are in outage.	114
5.6	Confusion matrices and reconstruction error plot of Autoencoder for the selected combination of MDT data density, and outage severity.	117
5.7	Confusion matrices of XGBoost for the selected combination of MDT data density, and outage severity.	118
5.8	Comparison of real and fake outage separation for MRFM using LOF. . .	121

List of Tables

2.1	Features helping human ear to differentiate the voice, generally used for clinical diagnosis.	14
2.2	Biosensing devices for data acquisition and their associated attributes. . .	18
2.3	Feature Engineering for building robust ML classifiers.	33
2.4	CNN-LSTM with Attention layer architecture.	38
2.5	Model performance metrics for cloud-based anomaly screening.	39
3.1	Distinctive features of SoTA literature for proactive pandemic prediction.	45
3.2	Wearable sensing devices and their applications in disease detection. . . .	50
3.3	List of symbols used to develop EpiTrack framework.	54
3.4	Simulation parameters for mobility and infection spread modeling.	63
3.5	Performance comparison of ML models for infection risk prediction. . . .	66
4.1	Comparison of state of the art literature and our work.	75
4.2	NAF/SON functions and the realted parameters.	80
4.3	Transition probabilities of NAF functions resulting in optimal or suboptimal network parameter configurations.	85
4.4	Discrete time markov chain (DTMC) states.	87
4.5	Mobile network setting with initial probabilities.	88
4.6	Description of the simulation parameters.	90
5.1	Description of simulation parameters.	108
5.2	Implication of MDT attacks on SON Functions.	109
5.3	Description of simulation parameters for outage scenario.	114
5.4	Performance comparison of the anomaly detection module for the Autoencoder and XGBoost.	119

Abstract

Ubiquitous connectivity, powered by mobile networks and smart digital technologies, is transforming critical sectors such as public health, industrial automation, and smart cities. However, the increasing complexity and interdependence of these connected systems present unprecedented challenges in maintaining performance, security, and reliability, particularly for mission-critical applications that demand ultra-reliable, low-latency communication. Traditional anomaly detection systems, which rely on reactive fault reporting, are no longer sufficient, as they often identify issues only after service degradation has occurred.

This dissertation addresses these limitations through four novel AI-driven anomaly detection frameworks that enable proactive, scalable, and context-aware intelligence by integrating emerging mobile networks, user mobility traces, and secure pervasive connectivity facilitated by smart devices. First, a hybridHealth AI framework is introduced as a smartphone-based screening tool that combines on-device acoustic analysis for rapid detection with deep learning-based detailed diagnostics in cloud environments, optimizing resource efficiency and ensuring high diagnostic accuracy and fairness, while addressing challenges such as data inconsistencies, label scarcity, and demographic biases. To advance screening beyond the individual level and enable population-scale epidemic hotspot prediction, EpiTrack is proposed as a novel outbreakprediction framework that leverages real-time user mobility traces from mobile networks to enable accurate and proactive public health interventions. This data-driven machine learning model offers unprecedented forecasting precision, helping prevent outbreaks from escalating into widespread epidemics.

In addition, smart-device-based screening tools and mobility-driven analytics rely heavily on continuous mobile network connectivity; any disruption directly undermines real-time anomaly detection and predictive intelligence. To maintain system reliability in anticipation of such disruptions, we propose a proactive anomaly management framework that models network reliability, predicts faults in advance, optimizes resource allocation, and

supports zero-touch automation in 6G and beyond networks. An additional barrier to achieving resilient, autonomous AI-driven systems lies in the vulnerabilities introduced by compromised or untrustworthy data collected through mobile networks and smart digital technologies. To address this challenge, we propose MRIF, a multimodule AI-based security framework that identifies and mitigates data-driven attacks, ensures accurate mobility insights, safeguards optimization processes, and reinforces trust in network intelligence.

By addressing the key challenges of fairness, proactivity, reliability, and security, this dissertation establish a unified, end-to-end ecosystem for AI-driven anomaly detection. Through this convergence of intelligent modeling and real-world integration, the work paves the way for next-generation autonomous systems that are resilient, secure, and significantly impact the public health outcomes worldwide.

CHAPTER 1

Introduction

1.1 Motivation

The rapid evolution of smart digital technologies and the exponential rise in connected devices have led to a paradigm shift in how we design and operate intelligent systems. As a result, 5G and the emerging landscape of 6G—has become central to enabling next-generation services in autonomous systems, industrial automation, and smart healthcare. The seamless integration of smart devices, ranging from wearable biosensors and IoT-based monitors to connected mobility platforms, is reshaping human-environment interactions. This transformation demands communication infrastructures that are not only high-speed and ultra-reliable but also resilient, adaptive, and intelligent. However, as digital ecosystems become increasingly complex and interdependent, ensuring uninterrupted connectivity and optimal network behavior has become profoundly challenging. Network congestion, parameter misconfiguration, unpredictable mobility patterns, and cybersecurity vulnerabilities now pose tangible risks to systems tasked with life-critical decisions. In use cases like remote diagnostics, autonomous driving, or emergency response coordination, even transient performance degradations can cascade into significant operational failures. Traditional anomaly detection methods are largely reactive—identifying faults only after service-level indicators such as latency, throughput, or Quality of Experience (QoE) have deteriorated. In the context of mission-critical 5G services, this delayed response is insufficient. These methods fall short of meeting the ultra-reliable, low-latency requirements expected from smart infrastructure. To bridge this gap, there is an urgent need for proactive anomaly detection frameworks—those that leverage predictive analytics, real-time monitoring, and AI-driven intelligence to detect and mitigate issues before they materialize into service disruptions.

Artificial intelligence, when fused with ubiquitous smart technologies, holds the promise of unlocking such predictive capability. However, its full potential is hindered by several foundational challenges. The real-world smart device data is often noisy, incomplete, or inconsistently labeled. This affects the accuracy and reliability of AI-based diagnostics and anomaly classifiers. Second, current self-optimizing networks (SON) are still primarily reactive in their operation. They detect and adjust to faults only after network performance has degraded, resulting in resource inefficiencies and a diminished user experience. The existing mobile network optimization mechanisms operate reactively, the delayed response leads to prolonged service disruptions and suboptimal resource utilization challenges that are crucial for mission-critical applications. Further complicating the landscape is the growing threat of data-centric security breaches. As we increasingly rely on mobility traces, crowdsourced sensor inputs, and decentralized AI models for real-time decision-making, the threat of adversarial attacks becomes non-negligible. Malicious manipulation of network telemetry or user-submitted data can not only destabilize system behavior but put public trust at risk.

Therefore, achieving resilient and intelligent digital infrastructure requires a unified, multi-domain anomaly detection paradigm, one that is **proactive, scalable, and secure by design**. This dissertation advances this vision by proposing novel AI-powered frameworks that span smart healthcare, mobile network reliability, epidemic prediction, and data security. By addressing anomaly detection from the lenses of fairness, proactivity, and robustness for adaptive intelligent systems in an increasingly connected world.

1.2 Problem Statements and Outline of the Proposed Solutions

In this dissertation, we identify and address key challenges related to anomaly detection in AI-enabled digital ecosystems. We present the identified problems (**P**) and the proposed solutions (**S**) for each challenge.

P1: Ubiquitous connectivity in mobile networks and advances in the sensing capabilities of smartphones and smart digital technology have enabled on-device health monitoring.

However, existing solutions lack reliability due to **noisy real-world data**, **scarcity of labeled data**, and **inconsistencies in biomarkers collected through smart devices**, making accurate screening challenging.

S1: We propose a hybrid AI framework that optimizes performance and resource utilization by leveraging edge computing for rapid preliminary screening, while complex analyses are offloaded to a high-performance computing environment. This dual-layered approach ensures generalization, scalability, and robustness, enabling efficient and accurate anomaly detection.

P2: Conventional outbreak forecasting models are often **static** and **suffer from inherent delays**, as they rely on **crude assumptions** and **historical demographic data**, making real-time and accurate outbreak prediction challenging. Epidemiologists typically depend on complex models built from multiple data sources, such as transport networks, demographic trends, and surveys to estimate outbreak spread. Conventional epidemiological models are further constrained by inconsistencies and fragmented data, limiting their ability to provide time-efficient and reliable assessments of emerging outbreaks.

S2: To address this challenge, we propose leveraging mobility data harnessed from cellular networks-a paradigm shift in epidemiology and integrating advancements in AI to develop highly accurate mobility prediction models. These models, in turn, enable the real-time detection of hotspots, facilitating proactive interventions to prevent outbreaks from escalating into epidemics.

P3: Similar to outdated anomaly detection in epidemiology, current cellular networks still rely on manual or **reactive fault detection methods**. The network automation frameworks-being reactive, are insufficient for meeting the ultra-reliability demands of emerging 5G and beyond (6G) use cases. Without proactive solutions, these networks risk falling short of the performance and **reliability expectations of mission-critical applications**.

S3: To address this extremely challenging problem we propose a predictive anomaly management framework that mathematically models network reliability, enabling early

detection of performance degradation. By identifying sensitive parameters and predicting fault transitions, the framework proactively mitigates anomalies, enhances network resilience, and ensures seamless connectivity in 5G and beyond systems.

P4: While predictive models improve fault anticipation, they cannot capture all anomalies, leaving systems dependent on real-time data, which itself can be compromised. Such **cybersecurity threats** pose a significant risk to mission-critical applications, including real-time outbreak monitoring, remote surgery, and autonomous systems, making them highly **vulnerable to disruption and performance degradation**.

S4: We propose an AI-driven security framework that proactively detects and mitigates adversarial threats using a multi-module architecture. By filtering malicious reports in mobile networks, it preserves data integrity against attacker manipulation and ensures reliable mobility insights and patient-related information for critical use cases such as outbreak monitoring and public health response.

1.3 Current and Planned Dissemination and Publications

Academic Awards:

- A1.** Awarded Gallogly College of Engineering Dissertation Excellence Award by the University of Oklahoma, Spring 2025.
- A2.** Fulbright Scholarship awarded by U.S. Department of State, Aug 2019-2024.
- A3.** National Merit Scholarship, University of Oklahoma, 2019-2024.
- A4.** General Electric innovate event nomination as distinctive PhD female and minority graduate student in STEM, Oct 2021.

Peer-Reviewed Journal Articles:

- J1. A. Ijaz,** M. Nabeel, U. Masood, M. Hashmi, I. Posokhova, A. Rizwan, A. Imran, Towards using cough for respiratory disease diagnosis by leveraging Artificial In-

- telligence: A survey,” in *Informatics in Medicine Unlocked*, 29 (2022): 100832, doi: 10.1016/j.imu.2021.100832.
- J2. A. Ijaz**, T. Saeed, I. Sadiq, H. N. Qureshi, A. Rizwan, A. Imran, “An AI-Enabled bias-free respiratory disease diagnosis model using cough audio,” in *Bioengineering*, 11(1), p.55, doi: 10.3390/bioengineering11010055.
- J3. A. Ijaz**, W. Raza, H. Farooq, M. Manalastas, and A. Imran. “An AI-enabled framework to defend ingenious MDT-based attacks on the emerging zero touch cellular networks.” in *IEEE Networks*, vol. 38, no. 1, pp.228-237, Mar 2023, doi: 10.1109/MNET.006.2100377.
- J4.** W. Raza, M. U. Farooq, **A. Ijaz**, M. Manalastas, A. Imran, “AI-powered resilience: A dual-approach for outage management in dense cellular networks,” in *Computer Communications*, p.108-129, 2025, doi: 10.1016/j.comcom.2025.108129.
- J5. A. Ijaz**, T. Saeed, M. shah, M. S. Riaz, A. Imran, “Epi-Track: An AI-Driven Framework for Proactive Epidemic Prediction,” submitted in *Informatics in Medicine Unlocked*, 2025.
- J6.** H. N. Qureshi, M. Manalastas, **A. Ijaz**, A. Imran, Y. Liu, MO. Al Kalaa, “Communication requirements in 5G-enabled healthcare applications: review and considerations,” in *Healthcare*, vol. 10, no. 2, p. 293. MDPI, 2022, doi: 10.3390/healthcare10020293.
- J7.** M. Shah, **A. Ijaz**, I. Posokhova, A. Imran, “The Role of Biosensing and AI in Diagnosing Bipolar and Borderline Personality Disorders: Systematic Review,” submitted to *Clinical eHealth*, 2025.
- J8.** M. Manalastas, M. Nabeel, **A. Ijaz**, S. M. Zaidi, U. Masood, H. N. Qureshi, H. Refai, A. Imran, “Design considerations and deployment challenges for TurboRAN 5G and beyond testbed.” in *IEEE Access* vol. 10, pp.39810-39824, Apr 2022, doi: 10.1109/ACCESS.2022.3166947.

- J9.** M. S. Riaz, M Shaukat, T. Saeed, **A. Ijaz**, H. N. Qureshi, I. Posokhova, I. Sadiq, A. Rizwan, A. Imran. “iPREDICT: AI enabled proactive pandemic prediction using biosensing wearable devices. Informatics,” in Informatics in Medicine Unlocked, vol. 46, pp. 101478, Jan 2024, doi: 10.1016/j.imu.2024.101478.
- J10.** **A. Ijaz**, H. Farooq, A. Imran, “On the Effect of Multi-Parameter Optimization on Network Reliability: A Stochastic Perspective,” submitted in Computer Networks and Communications, 2025. AI-powered resilience: A dual-approach for outage management in dense cellular networks
- J11.** **A. Ijaz**, T. Saeed, A. Imran, “CoughSense: A Semi-Supervised Framework for Class-Imbalanced Respiratory Disease Diagnosis,” (under internal review, for submission in IEEE Letters).
- J12.** **A. Ijaz**, T. Saeed, A. Imran, “SmartScreening: An AI-Powered Framework for Smart Screening and Digital Health Diagnostics,” (under internal review), 2025.

Peer-Reviewed Conference Papers:

- C1.** M. U. Farooq, M. Manalastas, W. Raza W, **A. Ijaz**, S. M. Zaidi, A. Abu-Dayya, and A. Imran, “Data driven optimization of inter-frequency mobility parameters for emerging multi-band networks,” In GLOBECOM 2020-2020 IEEE Global Communications Conference (pp. 1-6).
- C2.** M. Manalastas, M. U. Farooq, S. M. Zaidi, **A. Ijaz**, W. Raza, and A. Imran, “Machine learning-based handover failure prediction model for handover success rate improvement in 5G,” in IEEE 20th Consumer Communications Networking Conference (CCNC), pp. 684-685, Jan 2023.
- C3.** M. Nabeel, M. Manalastas, **A. Ijaz**, H. Refai, and A. Imran, ‘Investigating handover behavior with 5G and beyond TurboRAN testbed”, in Seventh International Conference On Mobile And Secure Services (MobiSecServ), pp. 1-6, 26 Feb 2022.

- C4.** W. Raza, M. U. Farooq, **A. Ijaz**, and A. Imran. “Data-driven intelligent outage management for high shadowing environments in 5G & B networks”, in IEEE International Conference on Smart Applications, Communications and Networking (SmartNets), pp. 1-6, May 28 2024.

1.4 Organization

This dissertation presents an end-to-end AI-powered framework for proactive anomaly detection and monitoring across multiple layers—individual health, population-level outbreak trends, and cellular network reliability—leveraging data from smart devices and mobile networks. Chapter 2 introduces a hybrid AI framework that addresses the challenges of data inconsistency, label scarcity, and noise in smart device-acquired biomarkers. The framework leverages edge-cloud collaboration to enable scalable, real-time screening and diagnosis. Chapter 3 presents EpiTrack, a smartphone-assisted framework that integrates AI models with real-time mobility data to predict potential infection hotspots and support proactive epidemic containment by leveraging mobility trace forecasting. Chapter 4 proposes a stochastic modeling framework based on Discrete-Time Markov Chains (DTMC) to proactively detect performance degradation in cellular networks, enabling early intervention and enhancing the reliability of AI-assisted digital health services. Chapter 5 introduces MRIF, a multi-module security framework designed to detect and mitigate adversarial attacks targeting mobility-driven data in mobile networks. This framework enhances the security and reliability of smart digital systems. Finally, Chapter 6 concludes the dissertation and outlines potential directions for future research to advance trustworthy, scalable, and intelligent AI-powered digital ecosystems.

CHAPTER 2

Addressing the Challenges in AI-powered Diagnostics through Resource-Aware Hybrid Computing

2.1 Introduction

2.1.1 *Motivation*

The unprecedented interest towards data-driven processes, intelligent software, and Artificial Intelligence (AI)-based tools has led to the development of smart online AI-aided platforms across various domains. These platforms offer dynamic, automated solutions for challenges in wireless networks, industrial automation, and healthcare, enabling real-time data analysis, decision-making, and system optimization. Due to huge improvements in computing power and storage over the last decade, AI has already gained substantial attention in many fields, including smart healthcare [1–3]. In response to the increased importance of using non-intrusive intelligent methods, and with the advent of AI, techniques have been evolved to become more refined and automated for the efficient information extraction. Artificial Intelligence aided solutions play a significant role to make the healthcare system more cost effective, personalized, precise, and proactive [4–6]. Moreover, the availability of AI solutions on smart online platforms further enhances accessibility, allowing end users to benefit from real-time insights, seamless integration with existing systems, and improved decision-making. These platforms bridge the gap between advanced technologies and everyday usability, empowering individuals and professionals alike to leverage AI-driven innovations effectively.

While AI algorithms have shown remarkable success in image-based diagnostics [7, 8], voice-based analysis is also gaining traction—driven by the rise of smart digital technologies—since acoustic signals often carry distinctive signatures of various health conditions [9–11]. For example, it has been shown that respiratory sounds, when analyzed by

leveraging ML or DL techniques, can provide significant insights, thus enabling a powerful diagnostic tool [12–16]. These AI models, integrated into smart platforms, demonstrate highly accurate and predictive outcomes, making them indispensable in revitalizing healthcare practices by enhancing efficiency, precision, and accessibility.

Among other respiratory biomarkers that can be collected using smart devices and platforms such as wheeze, crackles, breathing, and stridor, cough is one of the outstanding biomarker exhibiting unique features. Cough contains vital information for numerous respiratory conditions [17] manifesting as a symptom of over twenty medical conditions. The distinct latent features in cough can be exploited for the detection as well as preliminary diagnosis of various diseases by leveraging sophisticated and intelligent ML or DL algorithms trained through the cough acoustic data. These AI-based algorithms run directly on portable devices such as smartphones and smartwatches, hence triage screening of pulmonary diseases can be performed free of charge at home.

2.1.2 Related Work

Recent advances in AI and signal processing have catalyzed a growing body of research focused on automated diagnosis of respiratory conditions using biosignals. These efforts aim to enhance diagnostic accuracy and accessibility, particularly in resource-constrained or remote settings.

A systematic review for the computer-based analysis of lung auscultations and respiratory sounds is presented in [18]. In the literature, efforts have also been made in collecting and presenting works that deal with pulmonary diseases using AI [19, 20]. Kumar et al. [21] provided a brief review of DL classification models for the pulmonary diseases leveraging cough audio analysis, the authors also provided the diagnosis of ten respiratory conditions. The outbreak of ongoing COVID-19 pandemic steered the focus of research to investigate the potential of AI and the advances of ubiquitous computing for timely diagnosis and to combat the spread of the disease [22–30]. In recent years, extensive research has been done in this domain to develop the automated AI-based screening tools for the timely and

efficient detection/diagnosis of the respiratory conditions. Cough detection and diagnosis are essentially binary and multi-class classification problems, respectively, where the key features (shown in Figure 2.1) are used as inputs to the machine learning model, to learn the complex behavior of the cough audio signatures for the respiratory disease diagnosis. In Figure 2.1, we provide the categorization of the AI algorithms being implemented in the literature for the tasks of cough detection and disease diagnosis. The figure shows a range of different ML models 1) linear ML, 2) non-linear ML, 3) deep learning, 4) ensemble, and 5) other statistical methods that are leveraged for designing an ML-based disease diagnosis frameworks. linear ML methods algorithms, such as Linear Regression, assume the training data to be of a specific functional form with a fixed size of parameters, whereas Non-linear ML algorithms, such as k-Nearest Neighbors and Decision Tree, are free to assume any functional form of the training data. Ensemble methods are of two types: Bagging and Boosting, which further have several variants. Deep learning methods on the other hand, such as CNN and DNN, are universal learners. These algorithms are of significant value when a lot of training data is available, as they do not require manual feature engineering and are capable of latent feature extraction on their own. Some papers such as [31–34] exploited hybrid or ensemble-based methods. Other statistical methods (also listed in Figure 2.1) are also used in earlier works on cough detection and diagnosis [34–38]. Among the most common combinations of feature set and ML model is MFCCs + SVM and the neural networks, which have the potential to model and achieve high accuracy. The performance of implementing different combinations of features and AI models such as MFCC+SVM, STFT+SVM and STFT+CNN are also combinely studied in [39], and the authors demonstrated that the CNN has the highest performance.

2.1.3 Contributions

The contributions of this chapter can be summarized as follows:

- We propose a hybrid AI framework for proactive anomaly detection in smart health

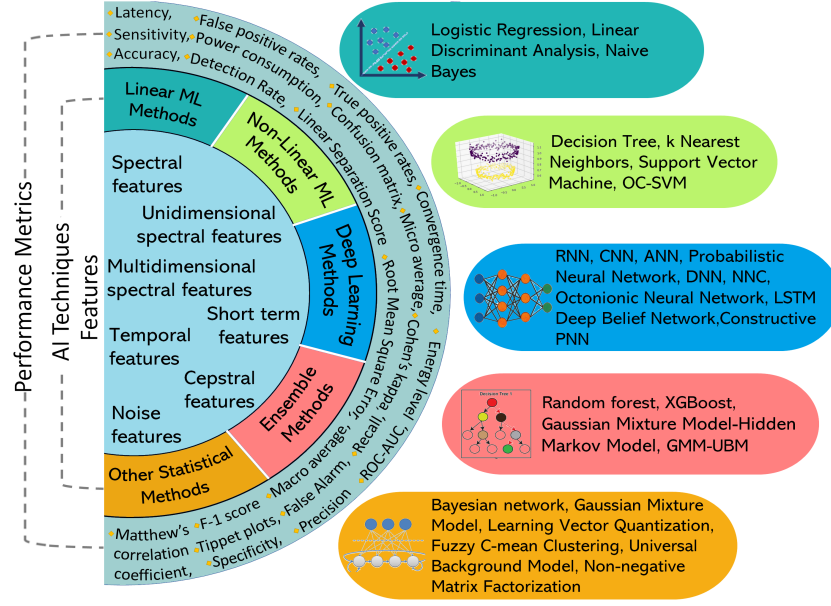


Fig. 2.1: Categorization of the AI techniques and performance evaluation parameters.

systems. The approach leverages edge devices for efficient on-device feature extraction and preliminary screening using lightweight machine learning models.

- To address the challenge of limited labeled data and imbalance in real-world audio datasets, we incorporate a semi-supervised learning scheme, enabling the framework to generalize well even under data scarcity.
- We deploy a cloud-based deep learning pipeline for high-resolution disease classification using engineered acoustic biomarkers, completing the edge-to-cloud workflow for reliable and scalable digital health screening.

2.1.4 Chapter Organization

The rest of the chapter is organized as follows. This chapter begins with the motivation, related work, and contributions (Section 2.1), followed by the diagnostic relevance of smart biomarkers (Section 2.2). Section 2.3 covers data acquisition and preprocessing methods for AI-based screening, while Section 2.4 highlights smartphone-based screening applications. Section 2.5 outlines key deployment challenges. Section 2.6 presents the proposed SmartScreening framework, and Section 2.7 presents performance evaluation. Finally, chapter conclusion is provided in Section 2.8.

2.2 Diagnostic Potential of Biomarkers Captured by Smart Technologies

In recent years, biomarkers have gained significant importance in detecting anomalies in an individual's overall health or lifestyle. These biomarkers can provide early warning signs for the onset of diseases, enabling timely interventions. For example, smartwatches now collect biomarkers like heart rate variability, blood oxygen levels, and physical activity patterns to detect conditions such as atrial fibrillation, hypertension, and sleep disorders, etc. Similarly, cough biomarkers are emerging as a critical tool for early detection and monitoring of respiratory diseases. These advancements underscore the transformative potential of integrating biomarker analysis with modern diagnostic tools.

Cough is considered as a symptom of various ailments. Experienced healthcare professionals use cough reflex interpretation as one of the manifestations helping them to associate it with discrete diseases. For instance, barking cough is associated with pertussis, whereas common causes of dry or hacking cough are upper airway cough syndrome, asthma, and gastroesophageal reflux disease. In case of viral croup, inflammation of the subglottic soft tissue occurs resulting in a rasping quality of the voice. Figure 2.2 shows a summary of cough types as well as different diseases associated to various human systems. Trained and experience physicians have been exploiting cough signatures to perform a differential diagnosis for various respiratory conditions, for instance, COPD, pneumonia, bronchitis, pertussis, pharyngitis, asthma, and tracheitis [17, 40, 41]. The reason for the accurate diagnosis by using cough is possible because the symptoms and the location of the underlying irritants in the respiratory system are quite distinctive, which lead to the audibly distinguishable cough sounds. Nonetheless, an untrained human ear is incapable of characterising the coughs caused by the different diseases delineated in the figure. It is evident from the figure that there is an overlap of cough types and the associated diseases, making it difficult for the physicians to use cough as a preliminary diagnostic tool with guaranteed accuracy.

For the characterization of cough based on acoustic analysis and auscultation, a human hear is able to distinguish cough based on limited audible features such as timbre,

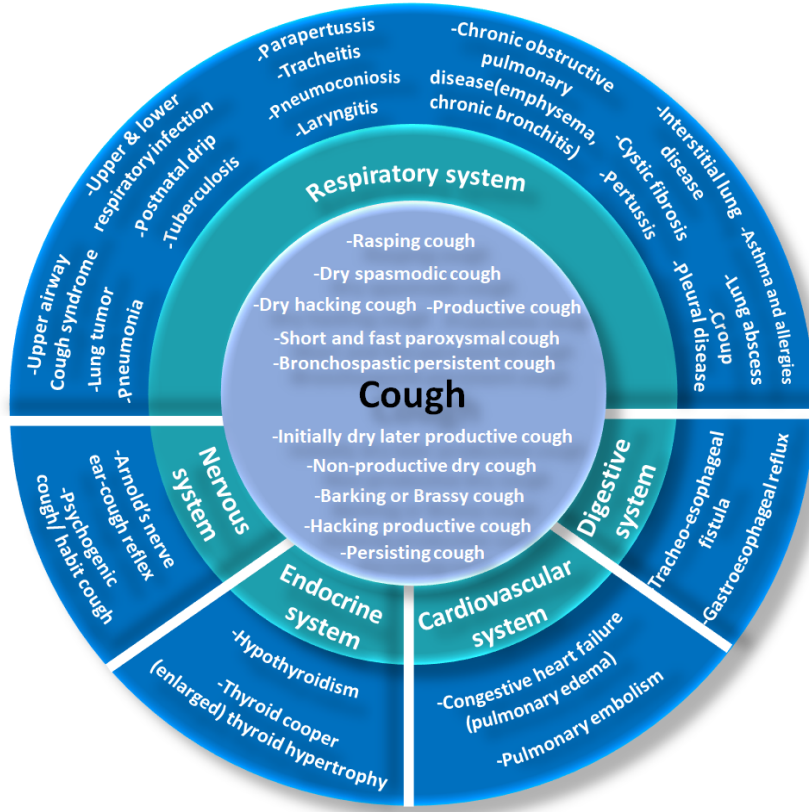


Fig. 2.2: Cough types and the associated diseases.

loudness, pitch, and duration. These different features along with their attributes and description are listed in Table 2.1. Other features that can aid the medical professionals to further classify the cough are wetness and dryness. Hence, common cough types that are considered as hallmark of certain diseases are identified based on the limited audible features. The fact that cough can be characterized in only five or six dimensional space by its virtue is limiting its ability to be uniquely associated with a certain disease. It is demonstrated in the study that dry or wet cough were precisely recognized in 76.1% of cases, whereas cough with accompanied wheeze was recognized only in 39.3% of cases, as it is difficult to distinguish wheeze in the cough [42]. Therefore, it can be interpreted that there is an overlap between the characteristics of different cough sounds, making it hard to be distinguishable by human ear based on the restricted features. Consequently, this is a dire need to complement the existing medical practices and to provide additional assistance to the physicians for better cough characterization and accurate disease diagnosis.

TABLE 2.1. Features helping human ear to differentiate the voice, generally used for clinical diagnosis.

Feature Name	Attribute	Description
Timbre	Harmonics, temporal	Multiple frequencies changing over time
Pitch	Psycho-acoustical (frequency)	Perception of high or low sound
Quality	Temporal	Sensation received by ear
Loudness	Amplitude	Intensity of a sound
Duration	Period	Length of time a pitch is sounded

2.2.1 *AI-Driven Biomarker Feature Analysis for Disease Screening*

Artificial Intelligence algorithms have the ability to project biomarkers associated with various diseases collected through the online platform in higher dimensional space and, hence are able to separate new unique features that are otherwise not possible to be distinguished by human ear due to the overlapping in the lower dimensional space [43–47]. This is because ML and DL models have the ability to process huge datasets and analyze hundreds of hidden features of coughs, which are beyond human capability to comprehend [48–51]. Thus, it is possible to develop a robust, reliable, and efficient AI-powered screening and diagnosis tool using cough sounds.

For instance, a human ear can detect only a handful of features (i.e., four to five). These features have binary possibilities such as dry or wet cough, barking or no barking cough, etc. Thus, there is a possibility of only $2^5 = 32$ combinations of features. In case of Artificial Intelligence, ML/DL models are able to classify numerous features, each feature further having a range of possible values. Theoretically, these models can classify between millions of possible combinations of features which human ear cannot distinguish. In case of cough audio, there exists $\prod_{i=1}^n c_i^m$ unique possible combinations of the resulting $m \times n$ feature matrix, where n corresponds to the number of total features, c_i corresponds to the number of possible values i^{th} feature and m corresponds to the number of audio subsegments in the whole audio segment. This means that even if we use one feature that can have only 5 possible values, then for 10 audio subsegments there still exist 5^{10} possible number of value combinations in the resulting 10×1 feature matrix. Thus, there is a high possibility that such a huge number of latent features would help in

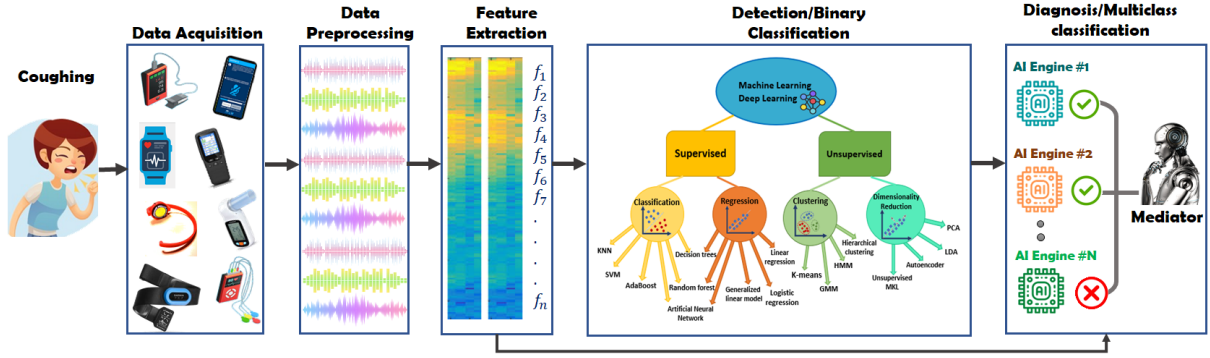


Fig. 2.3: AI-based cough detection/diagnosis process.

identifying the disease quite accurately and promptly. It is also well known that AI has the potential to differentiate subtle distinctions and nuances in the cough that are associated with the unique diseases. Hence, it can be exploited as a test medium for the diagnosis of diverse respiratory diseases. Several independent studies [51–53] have backed up this claim and demonstrated that the distinct cough latent features can be used for the accurate AI-based diagnosis of the respiratory diseases. To perform cough detection and preliminary disease diagnosis based on cough, we present a robust yet easy to develop AI engine in Figure 2.3. The major phases of a cough detection/diagnosis framework are data acquisition, data preprocessing, feature extraction, and selection of the appropriate machine learning models. These phases and their relevant characteristics are elaborated in the subsequent sections. The main task of the AI engine is to run a cough detection test and analyze whether the recorded sound is a cough or not. Further, it should be robust enough to detect cough in the presence of background noise. In case of diagnosis, the aim for this AI engine is to analyze cough and diagnose the correct disease accurately. By delegating most of the processing to the cloud; such an AI engine can be installed on portable devices. Thus, making it a resourceful screening tool easily accessible to patients for monitoring themselves and in some cases, providing preliminary diagnosis in order to encourage the patients to seek timely medical assistance. Such intelligent diagnostic tools can also aid physicians in better planning and providing care, ultimately leading to better outcomes and increased patient satisfaction.

2.3 AI-Enabled Biomarker Data Acquisition and Preliminary Screening

The training of an accurate, reliable, and comprehensive cough detection/diagnosis model requires the collection of representative and relevant data. Data collection step is the first and foremost step in the ML/DL classifier development process, as shown in Figure 2.3. This section presents a comprehensive overview of respiratory biosignal acquisition methods, with a focus on cough sound collection to support AI-driven health screening. It includes commonly used sensing modalities, public datasets, and demographic coverage, as well as key signal processing and feature extraction techniques. Together, these components form the foundation for building accurate and scalable diagnostic systems.

2.3.1 *Cough Sensing Modalities*

For building effective ML/DL-based diagnostic models, high-quality training data is essential—characterized by accurate labeling, low noise, and sufficient variability. The overall process of data collection generally includes acquisition, labeling, and refinement. As outlined in [54], data acquisition strategies typically fall into three categories: 1) Data discovery, involving the search for relevant public datasets; 2) Data augmentation, which enhances existing datasets through external sources or transformations; and 3) Data generation, including crowdsourcing and synthetic data creation when real-world data is scarce. Data acquisition is a fundamental bottleneck in AI-driven healthcare systems, primarily due to the reliance on trained medical professionals and the need for patient consent, especially during periods of illness. To alleviate this burden and support scalable solutions, smart healthcare sensors have emerged as critical enablers. These devices, equipped with embedded circuits, efficient computation, and long battery life, are designed to continuously capture health-related data without interrupting clinical workflows. In the context of cough-based diagnosis, the literature showcases a wide spectrum of data collection tools, ranging from basic voice recorders for simple acoustic capture [55], to sophisticated robotic systems engineered for automated collection and primary screening [56]. These biosensors not only gather audio but can also track en-

vironmental and physiological parameters, providing rich contextual signals around the patient. Table 2.2 summarizes several widely-used sensing devices for cough acquisition, highlighting key trade-offs such as ease of use, energy efficiency, noise sensitivity, and need for external assistance. Each wearable sensor or portable device has its own certain strengths and shortcomings. For instance, MP3 recorders are generally inexpensive, power efficient, and can tolerate high sound pressure levels, but are sensitive to high frequency and not immune to background noise [55]. Contrary to that, smartphones have the ability to capture continuous data, provide the dynamics of the real-life patterns, immunity to the background noise, readily available, user-friendly, and patient can place them within the close proximity. However, when the Signal to Noise Ratio (SNR) is low, the smartphone-based systems deactivate. Other challenges include the limited battery life and compatibility issues [57].

Cough collection using microphones and voice recorders is one of the most common practices in the literature [16, 31, 32, 36, 58–71]. The microphone can be placed near to the patient’s chest for several hours [58]. In [69] to suppress the surrounding noise while placing the microphone on the chest, the authors covered the microphone with a plastic foam membrane. The microphone can be attached to the shirt collar [31, 38] in home/office setting or lapel [65, 66, 72] for 24 hours while performing the daily routine tasks. Due to the technological advancements and ease of access, smartphones have become ubiquitous. Therefore, the researchers have also exploited the smartphone data recording feature to collect the cough data in [22, 37, 73–86]. Smartphones have noise robustness for up to -15 dB and can easily be placed in the patient’s pocket or the handbag for the data monitoring. However, as discussed earlier, the main challenge of using the smartphones for the data collection is limited battery life, the authors in [75] devise real-time cough detection algorithm that is power-efficient. Smartphone-based cough detection frameworks have the potential to accurately detect cough from the passively collected patient data and reduce the load of labeling the coughs manually. However, to make it easier for the patients to monitor their respiratory conditions and report about any medical anomaly accurately to the physicians and clinicians, further research is required in this domain.

TABLE 2.2. Biosensing devices for data acquisition and their associated attributes.

Sensing Devices	Ease of use	Expensive	External help	Noise sensitivity	LOS limitation	Energy efficient
Smartphone	● H	● M	● L	● L	● M	● L
Cough recording robot	● M	● H	● H	● M	● L	● M
Piezoelectric Sensor	● M	● L	● M	● L	● H	● H
AioCare Spirometer	● L	● M	● M	● L	● H	● H
Hull Automatic Cough Counter	● M	● M	● H	● H	● M	● H
MP3 recorder	● H	● L	● L	● M	● M	● H
Portable voice recorder and microphone	● H	● M	● L	● M	● M	● H
Smartphone customized App	● H	● M	● L	● M	● H	● L
Smartwatch	● H	● M	● M	● H	● H	● L
Computerized Data Acquisition System	● L	● M	● H	● M	● M	● H
VitaloJAK wearable microphone	● M	● M	● L	● M	● H	● H
Online platform	● H	● L	● L	● H	● H	● H
Microwave Doppler sensor	● L	● M	● H	● H	● H	● M
Cough detection camera	● L	● H	● H	● H	● L	● M

Note: **H**: High, **M**: Medium, **L**: Low

Using microphone is also a common practice to acquire cough data either in hospital, lab or normal settings [70, 87–92]. A high fidelity computerized data acquisition system which is comprised of low noise microphone, laptop, professional quality pre-amplifier, and A/D converter unit, is used for the sound signal acquisition in [93–96]. Other data collecting sensors include the FDA approved cough monitoring device VitaloJAK [97], smartwatch [83, 98], piezo sensor [39], public telephone hotline [99], hull automatic cough counter (HACC) [100], cough recording robot [56], microwave doppler sensor [101], Aio-Care spirometer [53], YouTube [102], cough detection camera [26], and stethoscopes [103].

2.3.2 Open Access Datasets and Demographics

Although AI-based cough analysis has gained significant attention, especially for respiratory disease diagnosis, the availability of open-source, well-annotated, and diverse cough datasets remains limited. The outbreak of the SARS-CoV-2 pandemic has accelerated the interest in audio biomarkers, particularly cough, motivating researchers across domains to contribute to public data repositories to enable scalable AI-based solutions. Robust machine learning models require large, diverse, and balanced datasets, making open-source data vital for reproducibility and generalizability in real-world deployments.

- i. The COUGHVID dataset is the largest expert-labeled cough datasets in existence, comprised of more than 20,000 crowdsourced cough recordings, which includes a variety of demographic diversity [104]. The team collected the dataset, filtered it, and then labeled more than 2,000 recordings with the help of the expert pulmonologists. The database can be exploited for a numerous cough acoustic classification tasks.
- ii. A public database Corswara comprised of approximately 1000 samples of respiratory sounds such as, cough, breath, and voice [105]. By recording nine different sounds from each patient, this cough repository helps to extract the multi-dimensional spectral and temporal features. Its aim is to complement the PCR-based COVID-19 diagnosis methods. The sound samples are collected via worldwide crowdsourcing using a website application. The database is still in progress with the objective of

better detection and quantification of the disease bio-markers with the help of sound acoustics.

- iii. Cohen-McFarlane and team introduced a database named as NoCoCoDa, where the authors collected the cough events through the public media interviews with COVID-19 patients [106]. The data repository has 73 cough events and it is created by the manual segmentation, extraction, and annotation of data. They also investigated the severe cases of COVID-19, where the cough can be productive, as mostly the patients have dry cough. It is worth mentioning that this database is not public, however, it can be available on request.
- iv. Another open source yet limited cough repository is developed by the independent AI researchers through website known as Virufy [107]. The dataset consists of only 16 patients (10 male and 6 female) with 7 COVID-19 PCR test positives while the remaining 9 are negative test patients. The biomarkers of the collected dataset are temperature, cough, shortness of breath, and glucose level.

Despite of all the success in the data collection, it is evident that further efforts are still required to make the data collection process more systematic and accessible for the training of robust models. In addition, for the results to be reproducible the dataset should be public, that can assist in the development of vigorous ML/DL algorithms for the accurate disease diagnosis.

Demographic Coverage

Cough datasets used for AI-based analysis have been collected from diverse geographic regions, including Bangladesh [35], India [25], South Africa [28], the United States [73], China [64, 67], and Brazil [28]. This global collection reflects the growing awareness of the need for inclusivity and population diversity in AI model training.

The datasets span across various age groups, from infants only a few months old to elderly individuals over 60 years of age [33]. In terms of environmental settings, data collection

has been conducted in clinical environments, controlled laboratories, and daily-life scenarios, resulting in recordings with varying noise levels and background interference. These variations mimic real-world use cases but also introduce challenges in generalization and noise robustness. In many studies, datasets include healthy volunteers alongside patients to train models that can distinguish between normal and pathological coughs [60, 108]. However, certain datasets focus solely on specific cohorts—e.g., children or elderly populations—limiting their broader applicability. A major concern lies in the imbalance of dataset size and class distribution. Some datasets contain only a handful of participants [63, 90], while others feature large-scale crowdsourced data involving thousands of users [24, 104]. Moreover, certain conditions (e.g., dry vs. wet cough, mild vs. severe infection) may be underrepresented, hindering the model’s ability to learn discriminative features for all categories. Finally, labeling inconsistencies—often due to subjective interpretation or lack of expert annotation—further complicate the training process and reduce the reliability of the resulting models.

2.3.3 Signal Refinement for Acoustic Analysis

The completion of the data acquisition step leads to the next crucial step within the data analytics pipeline, i.e., data annotation and pre-processing. This step has paramount importance as it is required for the refinement of the cough samples and for designing a high quality robust prediction model, as shown in the Figure 2.3. Data pre-processing helps to boost the consistency and accuracy in the collected data, thus providing ease in the data interpretability [109]. In general, there are four main steps in data pre-processing: 1) data cleansing, 2) data integration, 3) data reduction, and 4) data transformation [110]. During the pre-processing, the acoustic data is segmented into frames by isolating the part of the captured signal from the silence. It is a step that delineates the region of interest, i.e., cough events. For this step various tools and speech analysis software packages are used, for instance, Audacity, PRAAT [111], WEKA [83], Wavesurfer [112], and MATLAB Signal processing tool-box [100]. These tools are used to remove the silent portion at the beginning and end of the captured recordings, and for the rescaling and down-sampling.

PRAAT is also widely utilized for the data annotation [34, 64, 67, 73, 75, 111]. In order to successfully perform this step, the acoustic data is segmented into frames, RMS energy for the acquired frame is calculated and compared with a predetermined thresholds. In case the frames have low energy, it is assumed to be silence or ambient sound and, therefore, are discarded. High energy windows are selected for further processing and feature extraction [39]. To further remove the noise and dispose of the minor artifacts, filtration is applied. Finally, the data is normalized within a given range. The above mentioned pre-processing steps are considered very significant as they help in obtaining a well refined and transformed cough signals that are free from any ambient noise or silence. These pre-processed frames can then be used for the feature extraction and data analysis. Hence, this step is essential to improve the accuracy and performance of the prediction models.

2.3.4 Feature Engineering for AI-Based Screening

Cough has distinct latent features [113, 114]. These features with discriminating characteristics can be extracted by employing the pertinent signal processing and statistical cough sounds transformations generally referred as feature engineering; to be used for the training of a sophisticated AI engine. It is a process of extracting the most dominating and discriminating characteristics of an audio signal to obtain a suitable cough sound representation for the classification. To perform feature extraction, the cough waveform is modified to a relatively minimized datarate, for further processing and analysis. These features can be broadly categorized as time domain and frequency domain features. Some of the time-domain features that can be extracted are absolute mean, absolute median, standard deviation, skewness, kurtosis, and zero-crossing rate; and frequency-domain features such as spectral centroid, spectral roll-off, spectral variance, MFCC, and spectral chroma [83]. Spectral/Cepstral features which are based on the frequency and are obtained by converting the time domain signal using the frequency domain Fourier Transform.

Cepstral features are also widely used to separate the cough features from other sounds like laughing, speech, and background noise. In [16, 22, 31, 115, 116] the authors exploit commonly using cepstral features such as MFCC, LPCC, and GFCC. MFCCs can be obtained by frequency transform of the log spectrum and these features take account for the non-linear response of the audio spectrum [117]. LPCCs are an extension of the linear prediction via autoregressive modeling in the cepstral domain [118]. The underlying differences among these features lie in the frequency representation scale and the feature dimensionality depends on the value of inner parameters values [78]. Apart from the aforementioned multidimensional features, there are some unidimensional features which are leveraged in [78], for instance, Spectral Standard Deviation (SpecSD), Spectral Skewness (SpecSkew), Spectral Kurtosis (SpecKurto), and Spectral Peak Entropy (SpecPeakEn), to train the k-NN model and demonstrated satisfactory performance. To capture the long term dynamics of the cough sound signals, temporal features such as zero-crossing rate, energy of the signal and maximum amplitude can also be used for the model training [95].

There are several feature extraction libraries that can be leveraged to extract the aforementioned features [39]. Some of the frequently used libraries in the literature are: Hidden Markov Toolkit (HTK) [58, 66, 68, 99], Librosa [22, 25, 116], Taros-DSP to extract MFCC features, jMusic to extract the spectral features [83], RASTAMAT Matlab Toolkit [67], OpenSMILE [119], and others include Meyda, Aubio, and LibXtract. Feature extraction libraries have certain distinct characteristics, i.e., Librosa helps to extract high-level and low-level features, however, it does not cluster the features. For the real time feature extraction, Meyda and LibXtract are more appropriate libraries, while for high level features and segmentation, Aubio is the suitable choice. Combination of LibXtract and Marsyas can also be used for the data annotation and visualization of the audio cough features [120].

2.4 Smartphone-Based Biomarker Collection and Health Screening

The growing demand for accessible, data-driven healthcare solutions has led to the emergence of AI-enabled medical frameworks. These frameworks show great promise in reshaping traditional diagnostic practices by enabling patient-specific, intelligent screening systems. However, the effectiveness of machine learning (ML) and deep learning (DL) models critically depends on access to quality and diverse training data. In recent years, the development of lightweight, end-to-end smartphone applications has gained considerable attention for bridging this data gap. These apps offer an easy-to-use, low-cost, and scalable way to collect clinically relevant data—especially in resource-constrained or remote environments. Smartphones, being ubiquitous and sensor-rich, provide an ideal platform to capture real-life user behavior and health biomarkers. Figure 2.4 presents a timeline of customized cough-detection apps and websites developed over recent years. The following discussion elaborates on each application and its role in advancing AI-assisted respiratory diagnostics.

- *Mobicough*: A cost-effective mobile application is developed for the real-time detection and monitoring of the cough in [34]. For the data collection, an audio logging program is developed for the smartphones. Patients were asked to wear the ear-phone speakers while performing their routine works during 3-6 hours of the day. Features such as MFCC, ZCR, and entropy are then extracted and, later, based on these features, Gaussian Mixture Model (GMM) is trained for the modeling of cough sounds. For the modeling of background sounds such as noise and speech, Universal Background model (UBM) is employed.
- *Healthmode*: A smartphone customized application is designed for the continuous cough data collection and detection analysis using CNN [121]. The data is collected by using the phone’s internal microphone and the cough data is sent to the secure cloud server for further analysis. The performance of the devised technique is compared with the six commercially available cough monitors. The results in terms

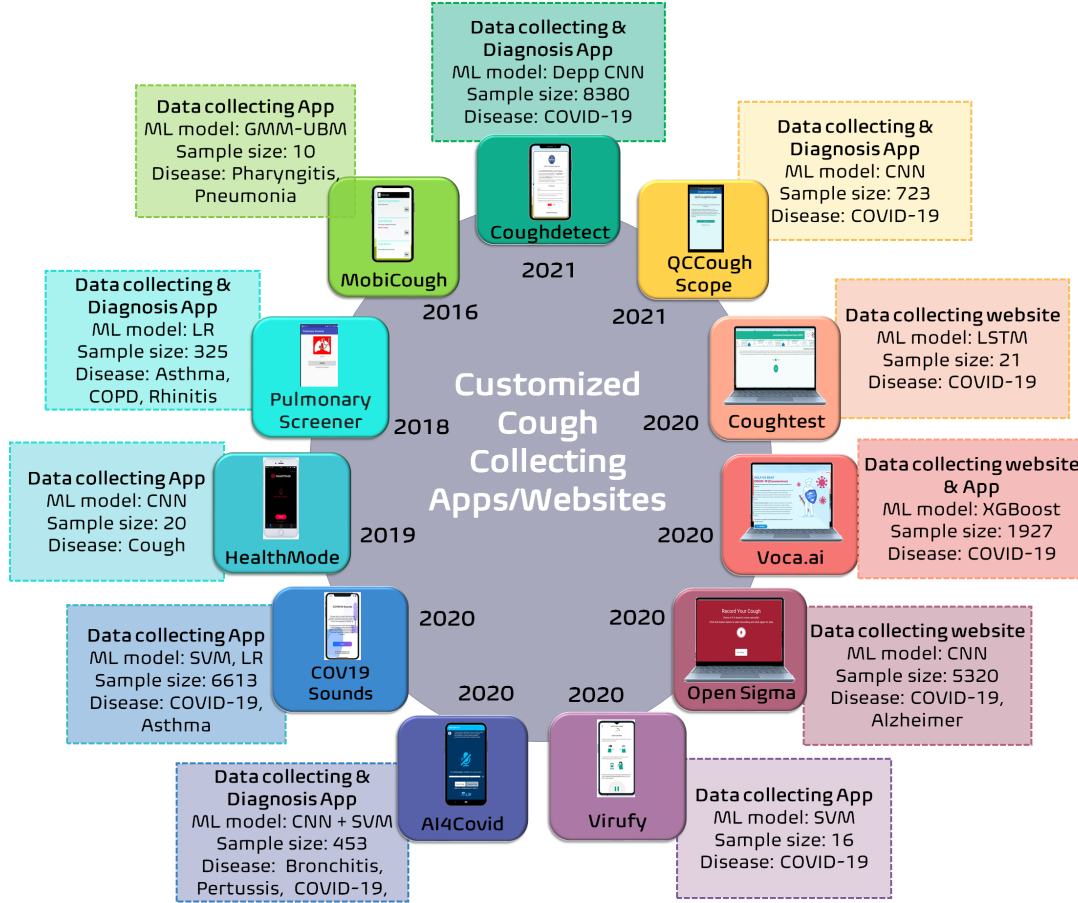


Fig. 2.4: Customized cough detection and disease diagnosing applications.

of sensitivity (90%), specificity (99.5%), and inter-rater agreement have showed comparable performance with the existing commercial solutions.

- *COVID-19 sounds*: Researchers in [24] has developed a cross-platform application for the collection of crowdsourced database of more than 10,000 cough samples from around 7,000 unique users till date. Out of this large dataset, 235 subjects are diagnosed with COVID-19, roughly comprised of 3.36% of the entire dataset. The authors assert that it is possible to classify COVID-19 patients from the healthy users, and the cough of COVID-19 patients from that of asthma cough with appreciable performance, even by leveraging simple ML models such as LR and SVM trained with the handcrafted features that are extracted using transfer learning. The proposed model is evaluated by using a subset of the datasets (141 cough and breathing samples). The models achieved an AUC (Area Under the Curve) of above 80% for all the tasks.

- *AI4COVID-19*: An application to capture the coughs of the patients suffering from bronchitis, pertussis, COVID-19 was developed by our team in [22]. In addition to collecting the cough data, this application proposed the diagnose of COVID-19 based on cough data. For the detection of coughs from other sounds, the authors trained a CNN model. While the diagnosis architecture leveraged a multi-pronged mediator centered risk-averse AI framework, showing an accuracy of more than 90% for the disease diagnosis. Transfer learning is leveraged to deal with the scarcity of the COVID-19 cough training data. This diagnosis framework is comprised of three independent classifiers, if the output of any two classifiers mismatches then the result turned out to be inconclusive.
- *Virufy*: An application and study developed by the researchers from 25 countries with an aim to help the low-income countries fighting against COVID-19 pandemic outbreak. The app encourages the people from eight countries to record their cough sounds using smartphones and help to train a robust AI algorithm for better cough pattern recognition. By using this Virufy dataset together with the other publicly available datasets, i.e., Coswara and Coughvid, the team in [122] developed a deep neural network to show the applicability of the crowd-sourced datasets for the detection of COVID-19 based on the cough data only. The trained AI-based method accurately predicted the COVID-19 infection with an ROC-AUC of 77.1%.
- *Coughtest*: In [28] the authors collected coughs from the patients who underwent SARS-CoV-2 virus test, using the application Coughtest [123] and combined data from the six continents. Before recording the cough, the subject has to answer the questionnaire prompted on the website regarding age, gender, country of residence, smoker or non-smoker, and some specific questions. After the normalization of data, features are extracted to train several classifiers; LR, SVM, MLP, LSTM, CNN, and a Resnet50. LSTM demonstrated the best performance in discriminating the COVID-19 positive coughs with an AUC of 0.94 based on 13 sequential forward search (SFS) best features.

- *Coughdetect*: A web-based automatic screening tool is developed for diagnosis of COVID-19 cough samples [124]. The subjects having symptoms such as cough, fever, shortness of breath etc. recorded their cough and sent to the server to have a preliminary diagnosis about the disease. The user of the app received an asynchronous message whether they should consult a physician. The authors extracted three sonographs namely MFCC, Melspec, and LPCS for the model training. In addition to cough, the authors used other biomarkers to predict the severity level of the disease. The data labeled based showed an encouraging AUC of 98.80% for the recognition of the three disease severity levels.
- *Opensigma*: A project by MIT researchers, aimed to diagnose COVID-19 by training the CNN model based on the cough data collected through website *opensigma.mit.edu*, which contains one of the largest COVID-19 cough datasets, comprised of 5,320 subjects. Transfer learning was applied for AI speech processing framework for the feature extraction. The CNN-based framework is trained with the data collected from the COVID-19 and Alzheimer’s patients. The trained model showed promising results in diagnosing COVID-19 with a sensitivity of 98.5% and a specificity of 94.2%.
- *Voca.ai*: To enable the swift diagnosis of COVID-19, Voca and researchers from Carnegie Mellon University worked on a project Voca.ai, which is an application and the website that asks the volunteers for the cough recordings [125]. The authors trained the XGBoost model using the crowdsourcing data with 1927 samples by extracting MFCC features to detect COVID-19 in the cough signatures of the healthy and unhealthy people. The performance analysis revealed that the model can effectively detect COVID-19 with an overall mean accuracy of 97%.
- *QUCoughScope*: Recently, an android application is developed by the researchers of University of Qatar to distinguish the COVID-19 patients. The data is comprised of the samples of 582 healthy individuals and 141 COVID-19 patients. The application asks for some symptoms-related questions and to record the audio signals. To

increase the sensitivity of the symptomatic COVID-19 patients and reduce the miss-classification, they deployed a cascaded pipeline. The overall sensitivity of the system reported by this study is 95.86%.

These customized cough detection frameworks offer the capability to passively monitor symptoms and collect data via smartphones or web interfaces. This reduces the dependency on medical personnel for manual labeling and enables scalable data gathering. As a result, AI-powered models built on this data can support preliminary diagnosis and assist clinicians in informed decision-making. However, to achieve real-time responsiveness and reliability at scale, there is a growing need to integrate both edge-level intelligence and cloud-assisted processing—motivating the development of generalized, scalable, cloud assisted frameworks.

2.5 Challenges in Developing Smart, AI-Driven Screening Systems

In the recent years, there has been much advancement in the domain of digital health technologies. The smart devices are being leveraged to collect the biomarker data and monitoring as well as preliminary screening of disease can happen on the devices by exploiting AI models or this data can be sent to cloud for further processing as discussed in previous subsection. However, to achieve this goal of digital disease screening using online platforms there are still many challenges that we have to deal with in terms of data, model, connectivity, security, and biases as listed in Figure 2.5. Some of the challenges are described as follows:

- *Data Fragmentation and Integration Challenges:* The healthcare ecosystem often suffers from fragmented and incomplete datasets, collected across different digital devices and online platforms in varying formats. This lack of uniformity complicates the integration of data for AI/ML models, limiting their ability to derive meaningful insights. Ensuring seamless data interoperability between smart devices, healthcare systems, and AI frameworks remains a persistent challenge.

- *Scarcity of High-Quality Labeled Data:* The availability of labeled data for training AI models is limited, particularly for niche applications like cough-based disease diagnosis. Furthermore, data imbalance, where certain classes are underrepresented, hinders the ability of models to generalize effectively. Crowdsourced data, while abundant, often lacks consistency, exacerbating this issue.
- *Noise and Artifacts in Data:* Data collected from smart devices often includes noise and artifacts caused by environmental conditions, user behavior, or device limitations. This can degrade the quality of the data, affecting the performance and accuracy of AI/ML models. Effective preprocessing techniques and robust feature extraction methods are critical to mitigating these challenges.
- *Hardware Constraints in Data Collection:* Smart devices while collecting data face critical hardware limitations, including limited battery life, restricted computational resources, and the risk of overheating during prolonged use. These constraints can impede their ability to reliably support AI-based diagnostic processes, particularly for real-time data analysis.
- *Algorithmic Bias and Model Reliability:* AI models can inadvertently inherit biases from training data, leading to skewed predictions that affect specific demographics. Ensuring fairness, and explainability in AI/ML systems is essential for building trust.
- *Scalability and Real-Time Processing Limitations:* Deploying AI/ML solutions at scale and enabling real-time analysis of large volumes of data is a significant technical challenge. Smart devices generate continuous streams of data that require computational resources, and optimized algorithms. Balancing real-time performance with accuracy is vital for successful implementation.
- *Connectivity Constraints in Data-Driven AI/ML Healthcare Technology:* Seamless 5G connectivity with high bandwidth and low latency is essential for data collection and for disease screening using AI/ML-powered frameworks. However, challenges

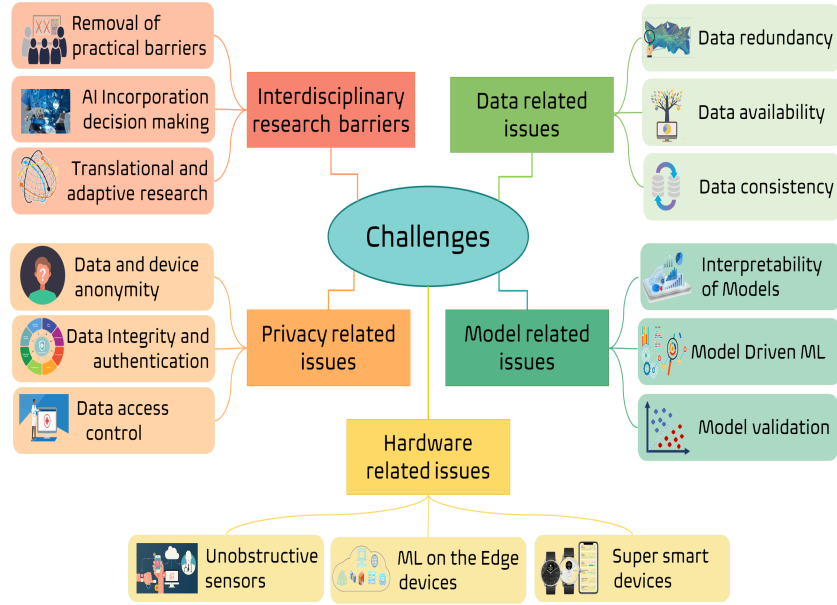


Fig. 2.5: Challenges faced during the incorporation of the AI techniques in healthcare domain.

such as inconsistent network performance, coverage gaps, and latency pose significant barriers to achieving reliable solutions.

2.6 SmartScreening: A Dual-Layered AI Framework for Smart and Reliable Diagnosis

To overcome the aforementioned challenges such as data fragmentation and integration, annotation delays, manual labeling, and latency in AI-based diagnostic systems, there is a need for an end-to-end, interconnected solution that ensures timely, accurate, and reliable healthcare support. Such solutions must optimize both real-time accessibility and computational efficiency. Emerging technologies—particularly edge/cloud computing and mobile 5G communication—offer promising capabilities for addressing these gaps [126]. By enabling seamless data transfer, localized inference, and scalable analytics, they can significantly enhance AI-powered diagnostics in digital health systems [127]. A comprehensive smart healthcare architecture is illustrated in Figure 2.6. The system begins with data acquisition via smart devices. Based on connectivity and available resources, data is either sent directly to the cloud for high-capacity analysis or processed locally on edge devices. This hybrid approach balances resource constraints with scalable and distributed

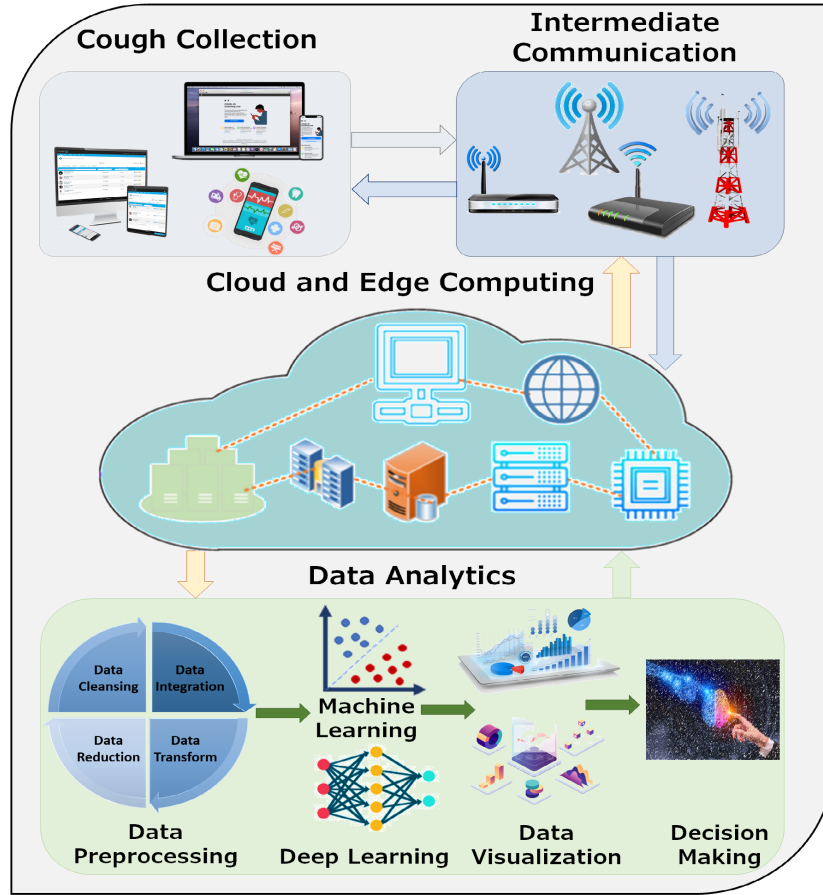


Fig. 2.6: SmartScreening: an AI-based hybrid respiratory disease screening framework.

solution.

In many real-world scenarios, particularly those in remote or resource-limited settings, it is not feasible to rely solely on centralized cloud infrastructure for screening and diagnosis. To address this, we propose SmartScreening—an AI-powered digital health framework that adapts based on infrastructure availability by integrating both edge and cloud computing environments.

Edge-Based Preliminary Screening: On-device preliminary screening is conducted using lightweight machine learning (ML) models such as XGBoost and Linear Regression etc. These models operate on domain-knowledge based extracted structured features (e.g., MFCCs, chroma, spectral signatures), offering high interpretability and low power consumption. Our proposed framework is capable to deal with another critical limitation in training AI systems, i.e., scarcity of publicly available annotated data. To address this,

SmartScreening integrates a semi-supervised learning component. This algorithm leverages both labeled and unlabeled samples to improve generalization, reduce dependency on manual labeling, and enable rapid deployment in edge devices. Thus, their efficiency makes them suitable for rapid preliminary health assessments in mobile environments.

Cloud-Based Deep Inference: For more complex inference tasks, the raw or minimally processed data is offloaded to the cloud where deep learning (DL) models such as Convolutional Neural Networks (CNNs), CNN-LSTMs, and CNN-LSTM-Attention architectures are employed. These models are adept at capturing spatiotemporal patterns in spectrograms and raw audio, automatically extracting high-level representations for robust disease classification. In addition, these are more generalizable and accurate model, extracting spatial and temporal entropy from cough acoustics for deeper analysis. The resources at the cloud provide scalability and computational depth for comprehensive diagnostics across large-scale datasets. *SmartScreening* offers several distinct advantages that make it well-suited for modern digital healthcare. It enables real-time, energy-efficient screening on smartphones through lightweight and interpretable machine learning models. For deeper and more complex analysis of unstructured biosignals, it leverages high-accuracy cloud-based deep learning pipelines. To address the challenge of limited labeled data, the framework incorporates semi-supervised learning techniques, enhancing diagnostic robustness. Furthermore, *SmartScreening* enables AI-powered disease screening that is not only accurate and efficient but also adaptable to both edge-level patient-centric environments and infrastructure-rich cloud deployments. A practical solution in diverse healthcare ecosystems—from individual smartphones to large clinical networks.

2.7 Results and Discussion

Following the data acquisition process, the recorded cough sounds underwent several pre-processing steps to ensure consistency and enhance signal quality. Initially, noise removal and silence trimming were performed using Audacity. Each mono-channel cough signal was sampled at 44.1 kHz and stored in PCM WAV format. Silent segments at the

TABLE 2.3. Feature Engineering for building robust ML classifiers.

Feature Name	Type	Number of Features
Mean Features	Spectral	12
MFCC	Spectral	20
Mel Spectrogram	Spectral	20
Chroma STFT	Spectral	12
Chroma CQ	Spectral	12
Chroma CENS	Spectral	12
Spectral Contrast	Spectral	7
Tonnetz	Spectral	6

beginning and end of recordings were clipped, and low-energy frames are associated with ambient background noise were discarded. High-energy frames were retained as potential cough events and normalized within a specific range. Only recordings with a duration longer than 2 seconds were considered for further analysis. These changes, though often imperceptible to the human ear, can be effectively captured through ML/DL models trained on structured audio representations. To train robust ML classifiers, a diverse set of spectral features were extracted from the cough audio data. These include Mel-Frequency Cepstral Coefficients (MFCC), spectral contrast, chroma features, and tonnetz, among others. Table 2.3 summarizes the extracted features and their respective types.

To further enhance classification accuracy and generalization, data augmentation techniques were applied to increase dataset diversity and address class imbalance. For structured features such as MFCCs and spectrograms, the Synthetic Minority Oversampling Technique (SMOTE) [128] was employed to generate synthetic samples for underrepresented classes. In addition, audio-specific augmentations such as frequency masking and time masking were applied to simulate varying real-world recording environments. These methods randomly suppress spectral or temporal components, introducing controlled variability. More advanced augmentation, such as SpecAugment [129], was also utilized, combining frequency and time masking with time warping to increase robustness against noise and signal variation.

Other techniques such as pitch shifting, speed perturbation, and addition of background noise further enriched the training dataset, making the ML/DL models more resilient

to real-world deployment conditions. These augmentations not only expand the training sample space but also simulate natural variability in cough events, which is critical for building generalizable and reliable AI-based diagnostic models.

To evaluate the performance of the proposed *SmartScreening* framework at the edge level, we leveraged supervised machine learning models such as XGBoost and Linear Regression are particularly well-suited for edge-level inference due to their efficiency, interpretability, and compatibility with structured features. XGBoost, a gradient boosting framework, provides fast and accurate predictions, making it ideal for resource-constrained environments like smartphones. It performs well on small datasets and is optimized for parallel execution, which aligns well with the needs of on-device screening applications. Linear Regression, while simpler, is highly interpretable and lightweight, making it a strong choice for real-time inference, especially when modeling linear relationships in early symptom assessments or initial triage. The XGBClassifier’s flexibility is further enhanced with hyperparameters such as `learning_rate`, `gamma`, and `colsample_bytree`, allowing for fine-tuned performance, especially in high-dimensional datasets.

Figure 2.7 presents a comprehensive comparative evaluation of two supervised machine learning models XGBoost and Linear Regression are applied to structured audio features extracted from cough sounds. This evaluation serves a dual purpose: (i) understanding the discriminative power of individual features through interpretability tools, and (ii) assessing model performance in the context of edge-based screening where computational efficiency and accuracy are critical. The SHAP summary plot on the left (Figure 2.7a) identifies the top 10 most influential features contributing to the prediction output of the XGBoost model. These include both handcrafted and domain-driven audio descriptors such as Mean Spectral Rolloff, Zero Crossing Rate, and Spectral Bandwidth, all of which capture key frequency and temporal properties of cough sounds. Interestingly, MFCC coefficients (e.g., $MFCC_3$, $MFCC_{11}$), MelSpectrogram components, and several Spectral Contrast subbands also emerge as dominant contributors. This highlights that both low-level energy-based features and high-level frequency distribution patterns are essential for robust cough classification. The SHAP plot reinforces the transparency and explainability

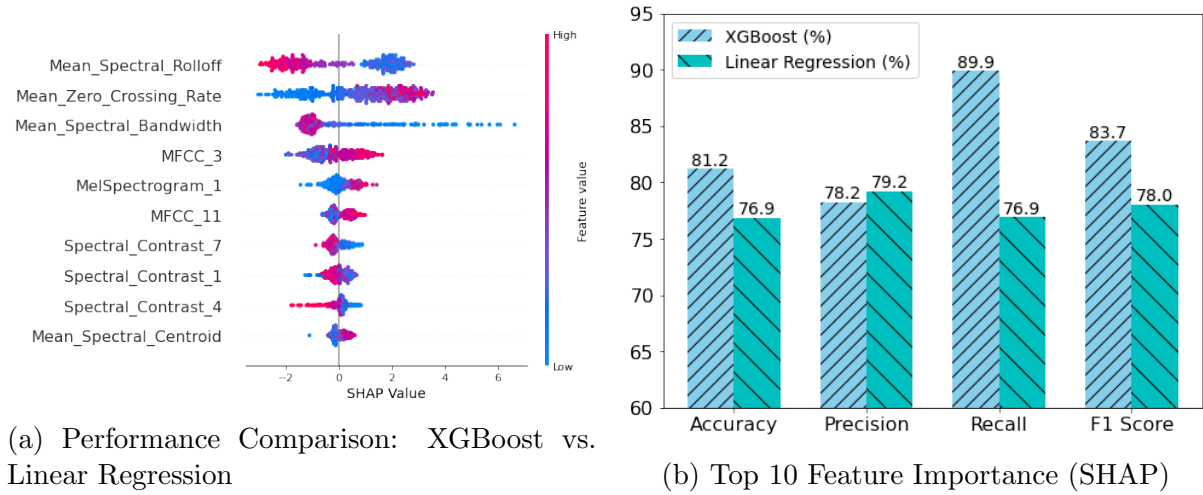


Fig. 2.7: Comparison of model performance and feature importance

of XGBoost, making it a suitable candidate for interpretable AI in healthcare applications. On the right (Figure 2.7b), the bar chart compares four core classification metrics, such as Accuracy, Precision, Recall, and F1-Score across XGBoost and Linear Regression. XGBoost demonstrates superior performance over Linear Regression across most evaluation metrics. It achieves a higher accuracy of 81.2% compared to 76.9%, indicating a greater overall rate of correct predictions. Although its precision (78.2%) is slightly lower than that of Linear Regression (79.2%), the difference is minimal and within acceptable bounds. Notably, XGBoost excels in recall, attaining 89.9% versus 76.9%, which reflects its enhanced ability to correctly identify true positive cases—crucial in medical screening scenarios where missing symptomatic patients can be costly. Finally, its F1-score of 83.7%, compared to 78.0%, confirms a better balance between precision and recall. These findings suggest that XGBoost learning strategy and regularization capabilities make it particularly well-suited for cough-based anomaly detection, especially when deployed on resource-constrained edge devices like smartphones. In contrast, while Linear Regression offers simplicity and speed, its performance limitations in capturing complex, non-linear relationships become evident in this task. Overall, the evaluation in Figure 2.7 provides a strong rationale for leveraging interpretable tree-based models like XGBoost at the edge level for real-time screening, while reserving more complex deep learning models for cloud-based inference pipelines.

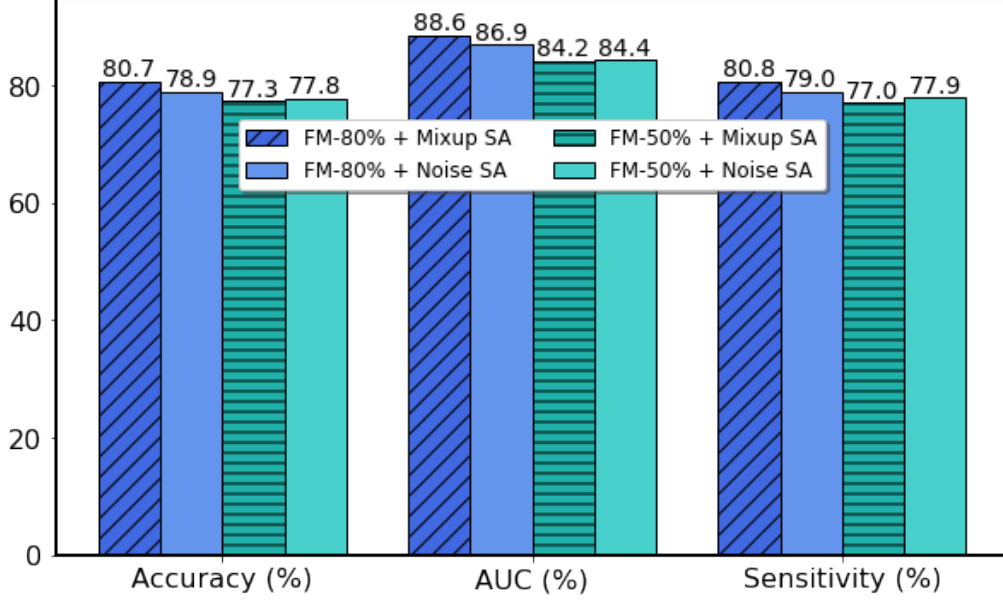


Fig. 2.8: Impact of semi-supervised training under varying label availability.

To overcome the challenge of limited labeled data in medical audio classification, we adopted the FixMatch algorithm, a semi-supervised learning (SSL) approach designed to leverage both labeled and unlabeled data effectively. FixMatch combines pseudo-labeling with consistency regularization by assigning pseudo-labels to weakly augmented unlabeled samples and enforcing consistency on their strongly augmented counterparts during training. In our experiments, we explored two FixMatch configurations using different proportions of labeled data: FM-80% and FM-50%. Additionally, we examined the impact of strong data augmentations including *Mixup* and *Noise* to improve generalization. Mixup interpolates between training examples to produce synthetic data, thereby improving robustness to small variations, while noise augmentation introduces perturbations that encourage the model to learn robust representations.

As illustrated in Figure 2.8, FM-80% with Mixup augmentation outperformed all other variants, achieving the highest AUC (88.6%) and sensitivity (80.8%). Even in the FM-50% configuration, applying Mixup and Noise helped maintain competitive performance. These results demonstrate that semi-supervised learning, combined with strong augmentation techniques, can significantly mitigate the impact of data scarcity in real-world medical applications, enabling effective model training even with reduced reliance on

annotated data.

Cloud infrastructure plays a pivotal role in enabling computationally intensive deep learning (DL) model training and large-scale data processing. This allows for seamless deployment and robust, real-time respiratory disease diagnosis. Within the proposed SmartScreening framework, cloud-based DL models such as CNN and CNN-LSTM are utilized to analyze spectrogram-based representations of cough biomarkers, offering high classification accuracy and adaptability for unstructured biosignal data. To enhance model generalization under label-scarce settings, the framework incorporates frequency and time-domain augmentation strategies. For instance, the CNN model used for disease screening efficiently extracts spatial features while minimizing overfitting. It consists of two convolutional layers (16 filters, kernel size (2, 2)), followed by ReLU activations, Average Pooling, and Dropout layers (rates of 0.2 and 0.5) for regularization. After feature extraction, a Flatten layer is applied before a dense layer with 32 neurons and a final sigmoid-activated output layer for binary classification.

To further improve anomaly detection performance, a CNN-LSTM model was trained. This model extends the simpler CNN architecture by incorporating multiple convolutional layers with increasing filter sizes (16, 32, 64, 128), batch normalization, and temporal modeling via an LSTM layer. The CNN-LSTM model reshapes the output into a time-series format suitable for sequential learning, with the LSTM capturing temporal dependencies. Dropout and pooling layers mitigate overfitting, while the model's depth improves its ability to learn complex feature hierarchies. The attention layer enhances interpretability and performance by assigning weights to the most relevant time steps. It computes alignment scores using a learnable weight matrix and softmax normalization to focus the model on important temporal segments. This context-aware weighting not only improves classification accuracy but also offers insight into the decision-making process that is critical component for trust in medical AI applications.

TABLE 2.4. CNN-LSTM with Attention layer architecture.

Layer	Details	Activation	Dropout Rate
Input Layer	Input shape = (Height, Width, Channels)	-	-
Conv2D + AvgPooling	16 filters, kernel=(2,2), stride=(1,1)	ReLU	0.2
Conv2D + AvgPooling	32 filters, kernel=(2,2), stride=(1,1)	ReLU	0.2
Conv2D + AvgPooling	64 filters, kernel=(2,2), stride=(1,1)	ReLU	0.2
Conv2D + AvgPooling	128 filters, kernel=(2,2), stride=(1,1)	ReLU	0.2
Reshape Layer	Converts feature maps to sequence	-	-
LSTM	256 units	Tanh	0.2
Attention Layer	Focuses on relevant temporal features	-	-
Dense	100 neurons	ReLU	0.5
Output Layer	1 neuron	Sigmoid	-
Optimizer	Adamax	Learning Rate = 0.001	-

Table 2.4 presents a comprehensive evaluation of multiple deep learning architectures, comparing their performance across two feature types: Mel-spectrograms and raw spectrograms. The CNN model using Mel-spectrograms achieved an accuracy of 0.80 ± 0.01 , with precision 0.83 ± 0.01 and F1 score 0.871 ± 0.01 . When switching to spectrogram inputs, however, the performance slightly degraded with lower precision and F1 score—suggesting that Mel-spectrograms retain more discriminative time-frequency characteristics suitable for CNN-based classification. Further improvements were observed with the CNN-LSTM model, which incorporates temporal context through LSTM layers. For Mel-spectrogram inputs, this model achieved 0.82 ± 0.01 accuracy and an F1 score of 0.87 ± 0.01 , showing better generalization than the baseline CNN. Notably, adding an Attention layer further

TABLE 2.5. Model performance metrics for cloud-based anomaly screening.

Model	Input Type	Accuracy	Precision	AUC	F1 Score
CNN	Mel-Spec	0.80	0.83	0.73	0.871
	Spectrogram	0.74	0.72	0.74	0.76
CNN-LSTM	Mel-Spec	0.82	0.86	0.76	0.87
CNN-LSTM	Spectrogram	0.76	0.74	0.76	0.77
CNN-LSTM (Attn)	Mel-Spec	0.83	0.88	0.79	0.88
CNN-LSTM (Attn)	Spectrogram	0.88	0.90	0.80	0.89

improved both precision and F1 scores. The CNN-LSTM (Attention) model using spectrograms outperformed all other configurations, reaching 0.88 ± 0.03 accuracy, 0.90 ± 0.05 precision, and a robust F1 score of 0.89 ± 0.03 , making it the most balanced model in terms of predictive capability and interpretability. Figure 2.9 provides additional insight into model behavior, particularly sensitivity and specificity—key metrics in diagnostic settings. As illustrated, Mel-spectrogram-based models like Mel-CNN and Mel-CNN-LSTM achieved high sensitivity (90.8% and 88.6%, respectively), essential for minimizing false negatives in disease screening. However, this came at the expense of specificity, which dropped as low as 55.7% in the case of Mel-CNN.

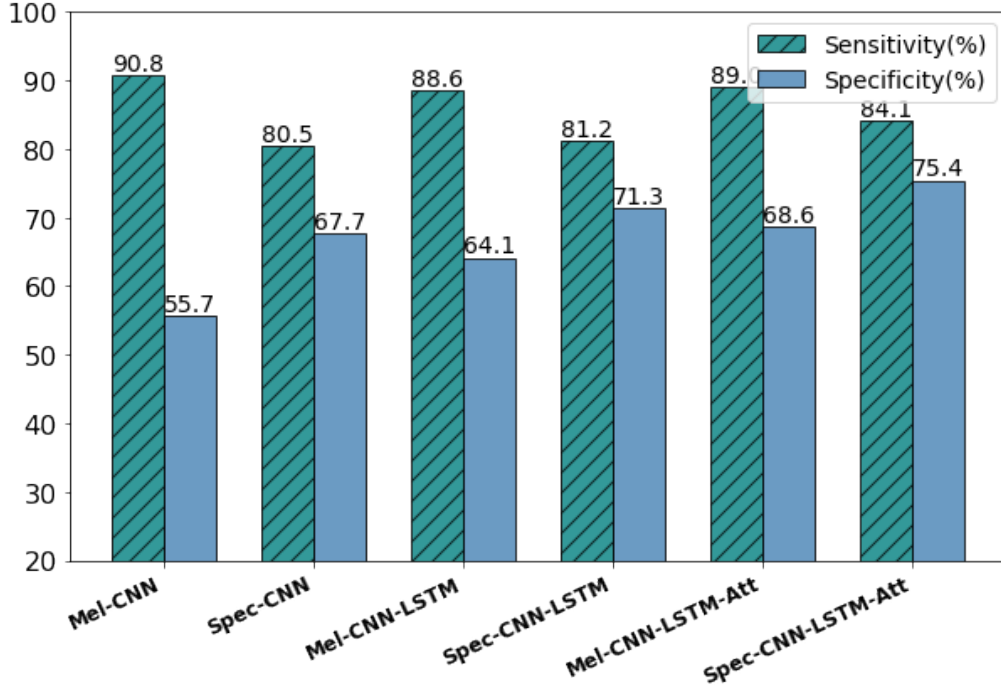


Fig. 2.9: Sensitivity and specificity comparison of deep learning models using Mel-spectrogram and spectrogram features.

In contrast, spectrogram-based architectures with attention (e.g., Spec-CNN-LSTM-Attention) offered a better trade-off, achieving 84.1% sensitivity and 75.4% specificity. These results highlight the attention layer’s ability to emphasize discriminative temporal patterns, making the model more adept at distinguishing subtle variations in cough sounds. This finding confirms that while Mel-based features amplify signal detail, spectrograms paired with deeper temporal architectures enhance robustness and classification balance. Overall, the results from Table 2.4 and Figure 2.9 support the use of hybrid deep learning models particularly CNN-LSTM with attention when aiming for high-fidelity diagnosis from complex biosignal data such as coughs. These models leverage both spatial and temporal cues to produce interpretable and reliable predictions.

2.8 Conclusion

In this chapter, we explored how digital technologies are transforming healthcare through AI-driven solutions for biomarker collection and analysis. We highlighted the role of smart devices in capturing cough biomarkers, emphasizing the importance of reliable connec-

tivity and domain-informed feature extraction. A hybrid AI framework was proposed, combining lightweight on-device machine learning for efficient preliminary screening with cloud-based deep learning for scalable, high-fidelity analysis of raw and complex data.

This framework addresses critical challenges such as limited edge device resources and the need for robust, interpretable feature engineering. By enabling seamless AI integration across edge and cloud layers, the proposed solution supports the development of an intelligent and connected healthcare ecosystem. The approaches presented demonstrate strong potential to improve clinical decision-making, enhance diagnostic precision, and accelerate the digitization of healthcare.

CHAPTER 3

EpiTrack: An AI-Driven Framework for Outbreak Forecasting by Leveraging Mobility Intelligence

3.1 Introduction

3.1.1 *Motivation*

Epidemics, especially those that escalate into pandemics, have historically resulted in catastrophic loss of life and economic disruption. Globally, these outbreaks have caused millions of deaths and trillions in economic losses [130–136]. In the United States alone, the social and economic consequences of the recent SARS-CoV-2 pandemic has surpassed the impact of all major natural disasters in the past century [137]. Despite progress in the predictive modeling of natural disasters, the global healthcare infrastructure revealed significant limitations in forecasting and containing disease outbreaks at their early stages.

Current epidemic forecasting methods largely rely on historical transport and demographic data, often requiring complex, high-latency modeling pipelines and manual data collection. These models tend to be reactive, suffer from inherent delays, and fall short in adapting to real-time population-level behavioral changes [137–146]. Meanwhile, inconsistencies in digitally acquired public health data and reporting lag further hinder timely, actionable insight.

To overcome these challenges, we introduce *EpiTrack* a novel, AI-powered epidemic hotspot prediction framework that marks a paradigm shift in digital epidemiology. EpiTrack departs from conventional demographic-based forecasting by leveraging data from mobile cellular networks and digital health diagnostics. Specifically, it uses real-time user mobility traces derived from cellular handovers and GPS signals, in conjunction with instantaneous diagnostic insights such as smartphone-based cough screening. This

combination enables highly accurate, location-aware epidemic forecasting with minimal infrastructure overhead.

The foundation of EpiTrack is built upon three pillars:

1. **Digital Testing on Smart Devices:** Advances in mobile health have enabled patients to self-administer tests and record diagnostic data (e.g., cough biomarkers) on smartphones.
2. **Real-time Result Aggregation via Wireless Connectivity:** Test outcomes are uploaded instantly to centralized systems via cellular networks, allowing immediate geo-tagging and analysis.
3. **Predictive Modeling of Outbreak Spread Using Cellular Mobility Traces:** Leveraging mobility traces, the system forecasts disease propagation patterns—identifying likely future hotspots at time t_2 based on diagnoses at time t_1 .

This mobile-centric model fundamentally redefines epidemic forecasting: instead of delayed lab confirmations and survey-based modeling, EpiTrack provides a low-latency, scalable, and highly adaptable digital epidemiology pipeline. The approach enables not only early detection but also anticipatory public health responses to avert epidemic escalation. Prior work has shown that prompt intervention significantly improves outbreak containment [147, 148], and EpiTrack operationalizes this insight through technological convergence of wireless connectivity, AI, and mobile diagnostics.

3.1.2 Related Work

Many studies have proposed solutions to proactively prevent epidemics, including analytical modeling based approaches, simulation based approaches and machine learning (ML) based methods, Table 3.1 describe the important feature extracted from literature. Analytical modeling based approaches consist of the works in [149]. The authors in [149] proposed a SI2C model. The model considered two types of human mobility, that included intra-region mobility, affecting local transmission speed, and inter-region mobility,

representing external infection sources. Using the proposed model, they analyzed disease spread with mobility data from 24 Chinese cities and 8 U.S. states. There are also several simulation based approaches, however, most of them use of them are based on analytical modeling, that rely on assumptions about the population, user behavior, and system parameters to maintain a tractable form. Some simulators like [150] do leverage a mobility simulator to simulate the population. However, it does not consider the impact of digital testing applications to prevent infection. Other approaches in literature consist of ML-based approaches, and continuous surveillance strategies. The work in [151] leverage an auto-reservoir neural network and dynamic network marker theory approach for the spread of COVID-19 to identify early warning of signals. Although the auto-reservoir neural network approach relies on smaller samples of data as compared to conventional forecasting algorithms like RNN and LSTM, and SVR, verified data on daily new cases is still required in a region to perform necessary correlations in the model.

Other approaches leveraging ML include anomaly detection based approaches, where anomaly detection is either performed on either patient health data (e.g., chest x-ray data) or on spatio-temporal daily new cases of epidemic [152, 152, 153]. However, patient health dataset is difficult to obtain due to patient privacy reasons, and might be very limited. Moreover, leveraging both patient dataset and spatio-temporal count of new cases results in a reactive approach to curb epidemic, rather than a proactive approach.

This chapter introduces a ubiquitously deployable framework that leverages smart digital technologies to monitor real-time mobility patterns for comprehensive disease surveillance. By capturing and analyzing mobility and behavioral data from devices such as smart-phones and network infrastructure, the framework enables proactive pandemic prediction without relying on manual intervention. It integrates low-cost, scalable components to detect early signs of both known and emerging pathogens, making it globally adaptable.

TABLE 3.1. Distinctive features of SoTA literature for proactive pandemic prediction.

Studies	Data Source	ML/Analytical Model	Predicted Parameter	Performance Metrics
Cui et al. 2022 [154]	U.S. Department of Transportation's Bureau	Exposure-risk model at the microscopic scale	No. of new infected cases	MAE, RMSE, and MAPE
Cui et al. 2023 [155]	U.S. Department of Transportation's Bureau	Linear regression & exposure risk with quality model	Average exposure risk and daily infection counts	Pearson correlation, obstacle size & placement
Tizzoni et al. 2021 [156]	Census surveys, mobile phone's billing information for Portugal, Spain, France	SIR model	Epidemic reproduction number, susceptible population	Jaccard similarity index, Spearman's rank
Schwabe et al. 2021 [157]	Telecom data from Swisscom	Poisson Model & Hawkes model	Prediction of infected cases	RMSE, MAE
Du et al. 2021 [149]	Baidu mobility data	SI2C model	New cases/day, cumulative cases	Pearson correlation, MAPE
Li et al. 2022 [158]	Subway swipe data from AFC system	SEIR model	Risk of infection	N/A
Chang et al. 2021 [159]	Facebook colocation and movement data	SLIR model	Risk of infection, regional import, case source	N/A

Studies	Data Source	ML/Analytical Model	Predicted Parameter	Performance Metrics
Fan et al. 2021 [160]	Mobile data by Verizon by 21 US counties	Ordinary least squares (OLS) regressions	Pandemic spread	radius of gyration, spatial census block groups
Li et al. 2022 [151]	COVID-19 datasets from Germany, Canada, Spain, Netherlands, Italy	Auto-reservoir NN	Daily new cases, Early warning signals	TPR, FPR, RMSE
Zhu et al. 2022 [161]	COVID-19 hotspots (CDC), Community mobility data (Google)	Gaussian process, Perceptron, Logistic, SVM, KNN, Decision tree	Hotspots probability, No. of cases, one-week ahead prediction	Precision, Recall, F1 score
Yabe et al. 2022 [162]	GPS traces, web search queries (Yahoo Japan)	Social contact index determination	Covid-19 cases, social contact index, high-risk user count, daily cases	N/A

Beyond enhancing outbreak response, this approach holds the potential to significantly reduce the socio-economic impact of future pandemics and catalyze innovation in public and animal health. Ultimately, it underscores the transformative role of smart technologies in closing critical gaps in modern epidemiological systems.

3.1.3 Contributions

The contributions of this chapter can be summarized as follows:

- Traditional human mobility models rely on extensive historical data (e.g., mobile phone records, GPS trails, transportation logs), which are difficult to acquire during the early stages of outbreaks. In contrast, our approach leverages realistic mobility traces, enabling timely and data-driven simulation of human interactions without the need for hard-to-access retrospective datasets.
- While existing works largely evaluate non-pharmaceutical (e.g., lockdowns) and pharmaceutical (e.g., vaccines) interventions, limited attention is paid to the role of digital testing and screening applications. To the best of our knowledge, this work is among the first to evaluate the impact of real-time, smartphone-based diagnostics in proactively controlling disease spread.
- Most analytical epidemic models depend on assumptions such as homogeneous mixing or perfect case reporting, which limit their robustness [163]. EpiTrack avoids such limitations through a simulation-based approach that models realistic agent-level mobility and interactions, offering improved scalability and reliability over closed-form epidemic forecasting.
- Unlike many methods that rely on detailed hospital and clinical datasets, *EpiTrack* is designed to function effectively even in low-resource settings. By utilizing mobile network-derived data and AI modeling, it offers early-warning capabilities where conventional health surveillance infrastructures may not exist.

3.1.4 Chapter Organization

Section 3.2 discusses the role of wearable devices in outbreak monitoring. Section 3.3 introduces the EpiTrack framework. Section 3.4 describes the simulation-based architecture, including mobility generation and infection modeling. Section 3.5 outlines the AI models used for prediction, including feature and graph based approaches. Performance evaluation and discussion is presented in Section 3.6. Finally, this chapter is concluded in Section 3.7.

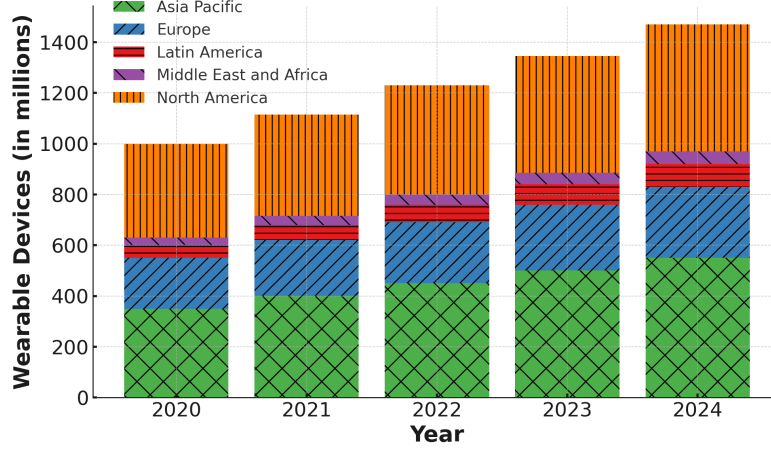


Fig. 3.1: Growth of wearable devices by region over the recent years.

3.2 Role of Wearable Devices in Biomarker Collection and Outbreak Monitoring

The rapid evolution of smart monitoring technologies over the past two decades has transformed healthcare by enabling non-invasive, continuous, and cost-effective surveillance. Smart devices—ranging from wearables and IoT-based systems to AI-powered sensors—offer powerful capabilities for individual health monitoring and large-scale outbreak prediction. Among these, wearable devices have gained significant traction, as shown in Figure 3.1, with over 1.4 billion units in use globally [164, 165], and healthcare-specific wearables accounting for approximately 350 to 490 million [166]. Wearables can be categorized into two main types: general-purpose devices that track parameters like heart rate and temperature, and disease-specific devices that monitor biomarkers such as glucose levels or respiratory indicators. While the former are widely available and consumer-ready, the latter demand advanced research and development to overcome technical limitations. Table 3.2 summarizes various wearable sensing devices and their applications in disease detection.

Beyond wearables, smart health ecosystems include mobile apps and non-invasive wireless sensors that utilize AI and big data analytics for health trend monitoring and early outbreak prediction. Figure 3.2 illustrates the proposed end-to-end system that begins with multi-modal data acquisition from devices like smartphones, smartwatches, and smart glasses, capturing physiological signals, mobility, and behavioral patterns. Collected data

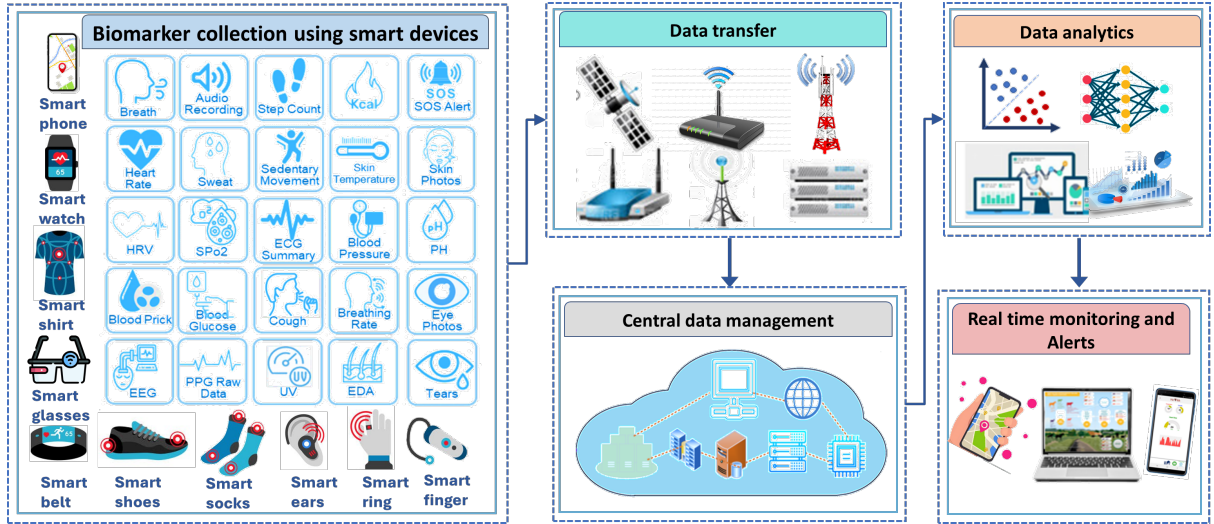


Fig. 3.2: Anomaly detection and outbreak prediction using biomarkers collected from smart devices.

is transmitted via 6G-enabled high-speed networks to the cloud for advanced analysis using AI-based models. These models identify health anomalies and potential outbreak indicators, feeding insights into analytics dashboards or smartphone applications. If abnormal patterns are detected, the system can trigger alerts and notifications, enabling timely intervention and informed decision-making. This integrated framework combining smart technologies, reliable connectivity, and AI-driven analysis presents a scalable, real-time solution for anomaly detection and epidemic forecasting. With the potential to support both individual and public health, it contributes significantly to global preparedness and healthcare resilience.

3.3 EpiTrack Framework

The widespread availability and computational capabilities of biosensor-enabled smartphones can be leveraged to track user mobility for outbreak surveillance. We propose the EpiTrack Framework, a comprehensive and data-driven system to model and predict the spread of infectious diseases and anomalies in real time, as shown in Figure 3.3. The framework operates on three core pillars: 1) Digital and on-the-go Testing via smartphones and updating infection status instantly to central database; 2) Real-Time, Location-tagged Result Collection through mobile networks; 3) AI-powered mobility pre-

TABLE 3.2. Wearable sensing devices and their applications in disease detection.

Wearable Sensing Devices	Electrophysiological Signs	Diseases Possibly Detected
Smartphones, Smartwatches, Fingertip Sensors	Heart Rate (HR), Heart Rate Variability (HRV)	Atrial Fibrillation, Myocardial Infarction, Generalized Tonic-Clonic (GTC) seizure, Stress, Neurological Disorders, Renal Failure
Smartphones combined with wearable ECG devices	Electrocardiogram (ECG)	Arrhythmia, Structural Heart Abnormality, Coronary Artery Disease
Smartphones combined with wearable headbands or headsets	Electroencephalogram (EEG)	Seizure Disorders, Sleep Disorders, Brain Tumors, Dementia, Stroke, Behavior Disorders
Wearable Sensors equipped in headbands, fingertip clips, and wristbands	Photoplethysmography (PPG)	Hypertension, Low Oxygen Saturation, Artery Diseases
Smartphones and Smartwatches	Blood Oxygen Saturation	Respiratory Distress, Pneumonia, Congestive Heart Failure, Renal Diseases, Pulmonary Embolism
Smartphones	Breathing Rate	Acute Hypoxemia, Asthma, Chronic Bronchitis
Smartphones, Smartwatches, and Earbuds	Inertial Sensors (Accelerometer and Gyroscope)	Parkinson's, Tremors and Seizures, Paralysis
Smartwatches (Optical Sensors)	Blood Pressure	Hypertension
Smartphones combined with wearable skin tags	Blood Glucose Levels	Diabetes Mellitus
Thermometers in wristbands	Skin Temperature	Diseases with fever-like symptoms
Smartphones (Cameras, IR, Depth Sensing, and IRIS scanners)	Biomedical Images	In-vivo diagnostics (Oral, Dermatological, Cervical, Eye, and Ear), Wound Recovery Monitoring, Microscopy, Thermal Imaging for Fever
Smartphones and Smartwatches	Audio Recordings	Cough Detection, Tuberculosis, Sars-Cov-2, Parkinson's, Asthma

diction to model the likely trajectory of an outbreak. This framework stands out due to its unique integration of input collection, mobility simulation, exposure estimation, and predictive modeling using AI/ML, all leading to actionable outputs for public health

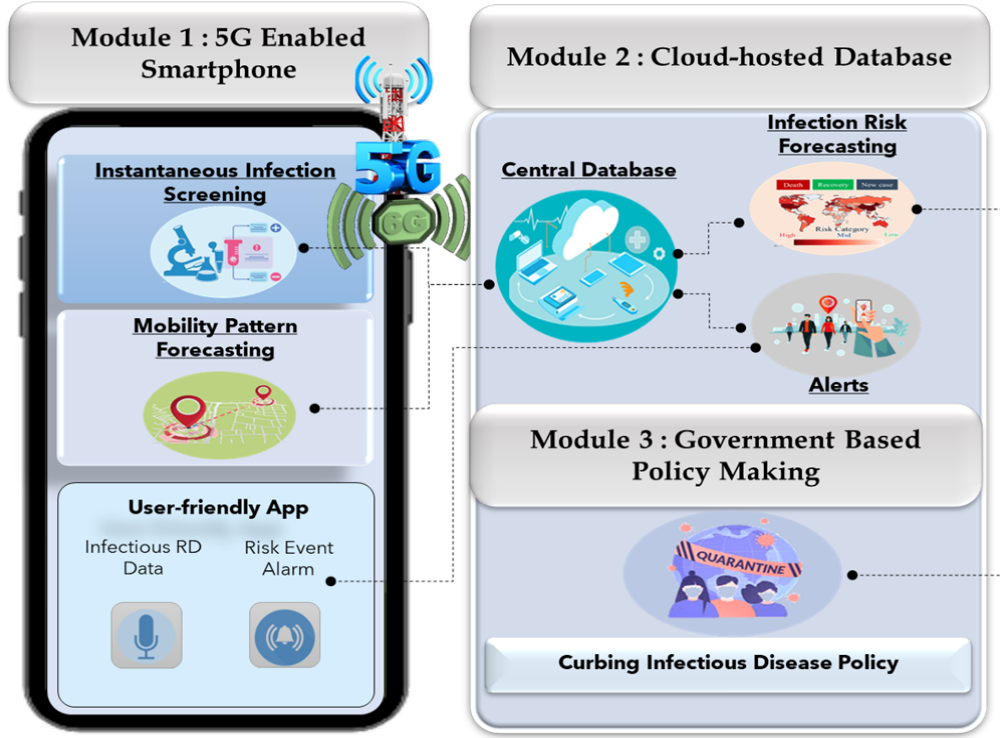


Fig. 3.3: EpiTrack : A mobile network-driven outbreak forecasting framework.

responses. The predicted insights can be used to proactively disseminate warnings and recommendations, preventing high-risk individuals from entering infected regions and thereby curbing spread.

The modular components of the EpiTrack Framework, as shown in Figure 3.4, illustrate a step-by-step flow from mobility trace acquisition to public health policy integration. The synergy among its components—from feature acquisition via smartphones to ML-driven exposure and transmission prediction—is pivotal in achieving accurate, timely, and scalable forecasting. The core stages of the framework are elaborated as follows:

1. **EpiTrack Input** The first step involves identifying critical features indicative of anomalous behavior or disease spread. Outbreak-prone diseases typically depend on environmental exposure and human interaction patterns. Features such as GPS-based mobility traces, inter-user distance, inter-contact time, and airborne droplet survival time become vital inputs. Environmental factors like temperature, humidity, airflow, and population density have been shown to significantly influence the transmission of respiratory viruses such as Influenza and SARS-CoV-2 [167, 168].

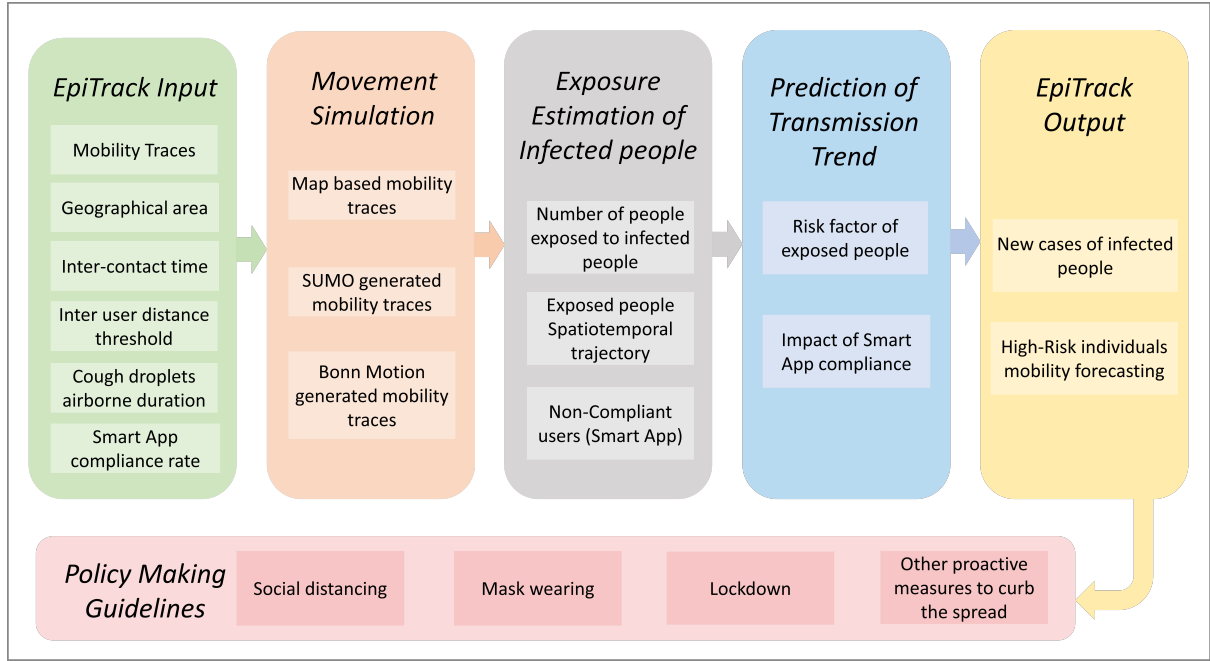


Fig. 3.4: EpiTrack framework with detailed modules.

For example, low humidity and poor ventilation are known to increase the risk of airborne transmission. Similarly, mobility and interaction dynamics, such as proximity within 2 meters for over 20 minutes [169], play an essential role in modeling disease exposure. These factors are encoded into the model for accurate downstream predictions.

2. **Movement Simulation** Once the key parameters are identified, the next stage involves simulating human mobility in realistic city settings. This is achieved using mobility simulators like SUMO (Simulation of Urban Mobility) and BonnMotion. SUMO enables vehicular and pedestrian trace generation from OpenStreetMap (OSM) data, simulating dense urban environments, while BonnMotion is particularly effective for wireless and ad hoc network mobility simulations [170]. These tools allow us to emulate varying population densities and user behavior, generating synthetic mobility data with high fidelity to real-world scenarios.
3. **Exposure Estimation of Infected Individuals.** The synthetic traces are processed through machine learning models to estimate the number and spatiotemporal spread of exposed individuals. This phase predicts when and how people are likely to come

into contact with infected users and detects non-compliant individuals—those not using the smart application or not adhering to mitigation protocols. By analyzing inter-user contact patterns, time spent in proximity, and mobility trajectories, the model estimates individual risk levels. Smart app compliance is also modeled to examine its effect on controlling transmission, demonstrating how digital interventions (like alerts and digital quarantines) reduce exposure likelihood.

4. **Transmission Prediction and Output Insights.** This predictive layer analyzes risk propagation across space and time, identifying potential future hotspots. The EpiTrack model outputs include newly exposed individuals, high-risk zones, and the projected movement of infected individuals. These outputs provide a basis for preemptive policy action. The model further integrates feedback mechanisms from user interaction, public health directives, and smart app compliance data to refine future predictions. The final output supports guidelines such as mask mandates, lockdowns, and social distancing policies, delivering real-time, data-backed recommendations to governments and healthcare agencies.

This section outlines the architecture of our simulation-driven framework, *EpiTrack*, which is designed to proactively model and predict the spread of infectious diseases based on proximity-based user interactions and temporal dynamics. The framework operates by simulating human mobility within a geographic area, determining exposure risk through spatial and temporal thresholds, and tracking infection transmission events over time. To facilitate clarity in understanding the simulation and modeling steps, Table 3.3 presents a comprehensive list of symbols and parameters used throughout the EpiTrack framework. These include variables representing mobility traces, inter-user distances, infection thresholds, simulation durations, and contact characteristics. The consistent use of these notations ensures precision in defining the underlying infection dynamics and the AI-driven predictive logic implemented in later sections.

TABLE 3.3. List of symbols used to develop EpiTrack framework.

Symbol	Description
N	Total number of users
T	Total simulation time
t	Sample time
n_{init}^d	Initial number of infected users
n_{end}^d	Infected at simulation end
r_{thr}	Radius of infection spread zone
d_u^t	Distance between u and infected user at t
t^d	Inter-contact time between users within r_{thr}
d_{thr}	Distance threshold for infection
d_{thr}^t	ICT (Inter-Contact Time) threshold
t_{thr}	Duration of airborne virus
x_u, y_u	X and Y coordinates of user u in area A
$M_u(t)$	Trajectory (mobility trace) of user u over time t

3.4 Simulation-Based Outbreak Prediction Architecture

3.4.1 Mobility Pattern Generation

For the generation of mobility traces we have done extensive review for the open-source simulator, and selected SUMO and BonnMotion based on the aspect that it can simulate the mobility patterns on the actual maps and follow the actual trends, this would help to analyze the realistic disease spread and to implement in real world maps. SUMO (Simulation of Urban MObility) is an open-source traffic simulation platform designed for network importing and demand modeling. SUMO requires a network file and a population definition file to generate mobility traces. The network file provides detailed information about roadways, intersections, sidewalks, and other areas where simulated pedestrians and vehicles can move. Roads and other details can be constructed or generated by converting OpenStreetMap (OSM) networks into SUMO-compatible formats, as depicted in Figure 3.5. The second required input is a population definition file, which outlines the demographics and activities within the network. This file includes details such as locations of residences, workplaces, schools, and the frequency of activities. In general, SUMO supports two pedestrian simulation models: the *Non-Interacting Model* and the

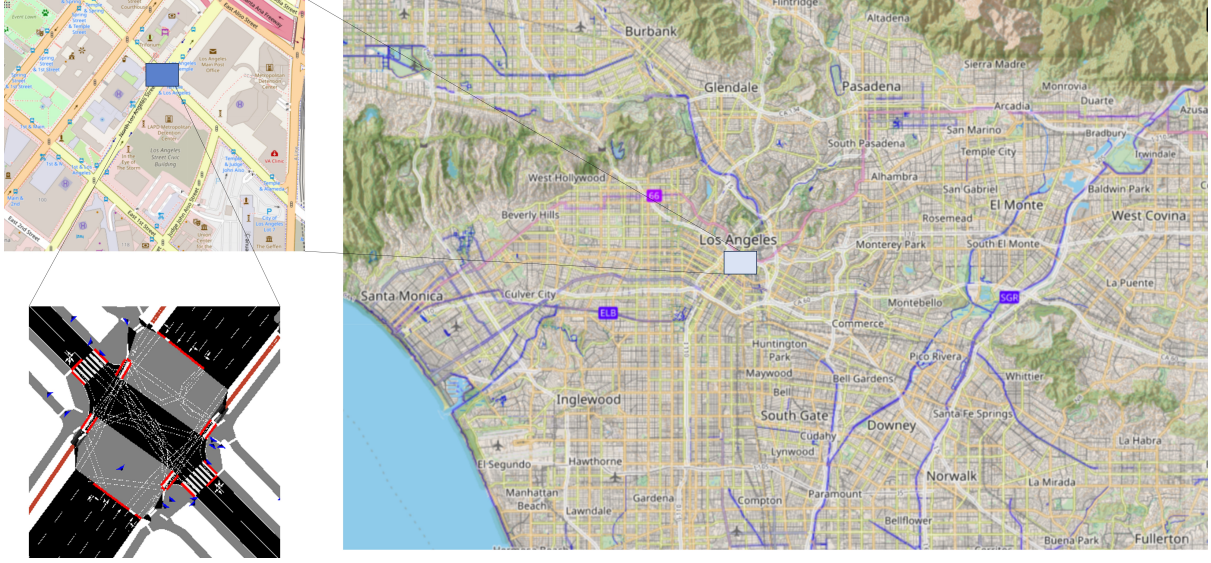


Fig. 3.5: Open street map and SUMO zoomed in road map in Los Angeles.

Striping Model, explained in detail as follows:

Non-Interacting Model: In this model, pedestrians travel at a constant speed without accounting for interactions with vehicles or other pedestrians. They "jump" across intersections and are free to move bidirectionally.

Striping Model: This is the default pedestrian model in SUMO. It uses 2D coordinates to define pedestrian positions within sidewalks, walking areas, or crossings. These coordinates, relative to the leftmost edge of a lane, are updated at each simulation step. Unlike vehicles, which primarily follow 1D movement, pedestrians exhibit more complex motion, advancing either along the natural direction of a lane or opposing it. The x-coordinate either increases or decreases monotonically as pedestrians move within a lane. Once they reach the end of a lane, they transition to the next, guided either by predefined paths or dynamic routing algorithms. For this Figure 3.5, we have selected a portion of map in Los Angeles as a network file and then provided a population definition file that contain all the necessary parameters in terms of pedestrian walking patterns, and speed etc.

BonnMotion supports various mobility models, including SLAW (Self-similar Least Action Walk), Random Waypoint, Gauss-Markov, Manhattan Grid, and Nomadic Mobility, allowing diverse movement patterns to be simulated. Users can fine-tune several parameters, such as node speed, pause time, waypoint distribution, movement constraints, and

trajectory smoothness, to replicate real-world mobility behavior. Additionally, BonnMotion enables customization of node density, simulation area, and duration, making it a flexible tool for evaluating network performance in dynamic environments. It offers the ability to export scenarios compatible with various network simulators and generate output files in CSV format, facilitating easy integration and processing within simulators. BonnMotion also supports map-based models to generate and analyze scenarios for performance evaluations. By leveraging OpenStreetMap (OSM) data, the tool integrates real-world maps into its environment, utilizing the OpenRouteService (ORS) interface to produce scenarios with realistic movement paths. The ORS interface employs an XML-based approach to define these paths. Specific movement patterns can also be created using arbitrary distributions to meet unique simulation requirements. While destinations are selected within a defined bounding box, routes may occasionally extend beyond its boundaries, especially when the "fastest" routing metric is applied. To address this, BonnMotion adds an extra margin around the bounding box to define the simulation area. Routes that extend outside this simulation area are deemed invalid and excluded from the analysis [171, 172]. This approach ensures that all simulated nodes remain within the designated simulation boundaries, enhancing the accuracy and reliability of mobility scenarios.

3.4.2 Modeling Infection Propagation Dynamics

The core objective of EpiTrack is to emulate realistic transmission dynamics using proximity and time-based interaction heuristics between users in a geographical region. EpiTrack is initialized with a set of key simulation parameters: the total number of users N , the initial infected user population n_{init}^d , the simulation duration T , and the spatial boundaries defined by area A . The framework is capable of ingesting mobility traces either synthetically generated (e.g., via SUMO or BonnMotion) or collected in real-time, along with additional metadata such as interaction duration thresholds, infection radii, and environment-aware modifiers. These parameters serve as the foundation for simu-

lating the disease transmission lifecycle across a population. The decision-making logic is summarized in Algorithm 1. At each simulation time step, the framework evaluates the infection spread based on spatiotemporal proximity between users. For every user u within the area A , the Euclidean distance d_u^t between u and any infected individual is first checked against a predefined infection threshold d_{thr} . This threshold defines the spatial zone within which infection is considered possible. If the user lies outside this threshold, no infection is transmitted and the simulation continues with the next interaction. If the distance condition is satisfied (i.e., $d_u^t < d_{thr}$), the algorithm proceeds to evaluate the interaction duration between the users. Specifically, the inter-contact time t^d (representing the length of the exposure) is compared with the temporal threshold t_{thr} . If the duration of contact is sufficiently long (i.e., $t^d < t_{thr}$), the infection is assumed to spread to user u , who is then marked as exposed. The infection calculations are repeated iteratively for all users within the area during each simulation step. The process terminates when $t_{sim} \geq T$, ensuring simulation stability and bounded execution.

This modular rule-based framework captures key epidemiological realities, such as the dual influence of proximity and exposure duration on infection transmission. Furthermore, it enables the incorporation of configurable disease-specific parameters (e.g., droplet survival time t_d^{air} or infection radius r_{thr}), making it adaptable to a range of infectious diseases beyond COVID-19. The output of the simulation includes lists of newly exposed users and their high-risk mobility paths. This flexible, rule-driven simulation can be extended for various diseases and integrated with behavioral data for better forecasting and early warning systems. In summary, EpiTrack provides a flexible, rule-driven foundation for understanding disease spread dynamics in mobile populations. Its strength lies in its simplicity, modularity, and ability to fuse real-world behavioral patterns with domain-aware simulation logic. When used in conjunction with AI-based predictive models and mobile sensing technologies, EpiTrack becomes a powerful tool for proactive and scalable public health surveillance.

Algorithm 1 EpiTrack Infection Spread Prediction

Input : Total number of users N , Initial infected users n_{init}^d ,
Geographical area A , Infection radius r_{thr} , Simulation time T
Distance threshold d_{thr} , ICT threshold t_{thr}

Output: Infection spread prediction, exposed users n_{end}^d , high-risk user trajectories $M_u(t)$

foreach user u in area A **do**
 Track coordinates (x_u, y_u) and mobility trace $M_u(t)$
 foreach infected user i **do**
 if $d_u^t < d_{thr}$ **and** $t^d < t_{thr}$ **then**
 └ Mark u as *exposed*
 else
 └ No infection spread
 └

while $t_{sim} < T$ **do**
 └ Repeat infection spread calculation for updated user positions and interactions

return Predicted exposed users n_{end}^d , infection chains, mobility trajectories $M_u(t)$

3.5 Implementation of Proactive AI Prediction Models

Artificial intelligence (AI) plays a pivotal role in the EpiTrack framework by enabling proactive outbreak forecasting through real-time analysis of mobility patterns, contact interactions, and infection history. The system integrates diverse learning paradigms tailored to multiple input modalities, including structured features, spatiotemporal trajectories, and dynamic contact networks. Classical models like Random Forest (RF) offer interpretable, feature-driven predictions, while sequential models such as LSTM and BiLSTM capture temporal patterns of exposure and infection progression. To model the irregular and non-linear nature of social interactions, EpiTrack leverages Graph Attention Networks (GATs) and Positional Transformers for direct learning from graph-based representations. These models are further extended to forecast future hotspot mobility using LSTM-based trajectory prediction. Together, they form a comprehensive suite of predictive tools supporting early warnings, dynamic risk assessments, and data-driven intervention strategies.

3.5.1 Feature-Based Learning for Exposure and Mobility Forecasting

Random Forest (RF)

Random Forest is an ensemble learning method composed of multiple decision trees trained on bootstrapped subsets of the data. It performs classification by aggregating predictions through majority voting, which enhances stability and reduces overfitting. RF is particularly useful for structured tabular data consisting of hand-engineered features such as inter-contact duration, contact frequency, and mobility density. In EpiTrack, RF is used to identify users at high risk of infection based on mobility logs and environmental exposure markers. Key metrics, such as feature importance, are derived using impurity-based or permutation methods to identify influential predictors in outbreak spread. For instance, features such as x_u, y_u (user location coordinates), number of close contacts, and local population density are often dominant.

Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks are a specialized form of Recurrent Neural Networks (RNNs), specifically designed to capture long-term dependencies in sequential data, which is crucial in outbreak prediction where user interactions and exposure events evolve over time [173]. Long Short-Term Memory (LSTM) networks are widely used for modeling sequential dependencies. Specifically, the *retention*, *update*, and *exposure* gates govern how much of the past memory is kept, how much new information is added, and how much of the memory is revealed to the output, respectively. This gated structure allows LSTMs to mitigate vanishing gradient issues and capture long-term dependencies in time-series data. This enables the network to retain and forget information selectively, making it ideal for processing user mobility sequences and predicting infection progression. Within the EpiTrack framework, LSTM is employed to model two key components: (i) exposure estimation based on past contact and interaction durations, and (ii) mobility forecasting for identifying high-risk regions based on infected users' future

movement. Given a time series of features such as location coordinates, inter-contact time, and mobility frequency, LSTM predicts both the likelihood of infection and the expected spatio-temporal trajectory of users.

$$z_t = \sigma(W_f u_t + V_f s_{t-1} + b_f) \quad (3.1)$$

$$r_t = \sigma(W_i u_t + V_i s_{t-1} + b_i) \quad (3.2)$$

$$\tilde{m}_t = \tanh(W_c u_t + V_c s_{t-1} + b_c) \quad (3.3)$$

$$m_t = z_t \odot m_{t-1} + r_t \odot \tilde{m}_t \quad (3.4)$$

$$e_t = \sigma(W_o u_t + V_o s_{t-1} + b_o) \quad (3.5)$$

$$s_t = e_t \odot \tanh(m_t) \quad (3.6)$$

Here, u_t denotes the input feature vector at time t , which includes location coordinates (x_u, y_u) , number of contacts, and exposure time for each user. The hidden state s_t and memory cell m_t capture evolving temporal dependencies. The gating variables z_t , r_t , and e_t serve as forget, input, and output gates, respectively, regulating what information to retain, update, or emit. Through this formulation, LSTM in *EpiTrack* dynamically learns from past exposure events and movement patterns, enabling the system to not only predict infection probability but also proactively estimate hotspot emergence based on future mobility of exposed users.

Bidirectional LSTM (BiLSTM)

While standard LSTM processes sequences in a forward temporal direction, Bidirectional LSTM (BiLSTM) extends this by learning from both past and future contexts simultaneously. This dual perspective is particularly advantageous in *EpiTrack*, where infection risk may depend not only on prior contacts but also on upcoming mobility behavior. BiLSTM consists of two parallel LSTM layers: a forward layer processing the sequence from $t = 1$ to T , and a backward layer processing from $t = T$ to 1. At each time step t , a bidirectional LSTM processes the input sequence in both forward and backward directions. The outputs from both passes are combined as:

$$s_t = [\vec{s}_t; \overleftarrow{s}_t] \quad (3.7)$$

Here, $\vec{s}_t$ and $\overleftarrow{s}_t$ denote the forward and backward hidden states at time t , respectively. Concatenating these two vectors enables the model to capture information from both past and future context, enriching the temporal representation, thus helping to better forecast infection spread.

3.5.2 *Graph-Based Learning for Infection Dynamics*

Traditional epidemic prediction approaches commonly utilize handcrafted features like mobility frequency, population density, and contact duration. While informative, such features sometimes may not fully capture the intricate and evolving patterns of disease spread, especially in densely connected or dynamically changing environments. In such settings, learning from interaction patterns directly without depending on explicit features becomes crucial. Graph-based learning, especially Graph Attention Networks (GAT), provides a powerful alternative. Graph-based models are better suited for scenarios where individual mobility and interactions form a dynamic network, a scenario where traditional tabular models may struggle to generalize. GAT can directly learn from how people interact (contact patterns), critical in dense environments or evolving outbreak scenarios.

Graph Attention Networks (GAT)

GAT is an advanced extension of Graph Neural Networks (GNNs) that introduces an attention mechanism to weigh the relative importance of neighboring nodes during feature aggregation [174]. Unlike standard GNNs that treat all neighbors equally, GAT dynamically learns attention weights based on the structural context of each node. In the context of EpiTrack, the population is represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ are users (nodes) and $\mathcal{E}$ are contact-based edges derived from proximity interactions. Each edge encodes a potential transmission pathway. GAT enables the model to focus

on highly influential individuals—those acting as super-spreaders—while diminishing the influence of less relevant interactions. Given a node i and one of its neighbors j , the attention coefficient α_{ij} , representing the importance of node j to node i , is calculated as:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}\mathbf{h}_i \parallel \mathbf{W}\mathbf{h}_j]))}{\sum_{k \in \mathcal{N}_i} \exp(\text{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}\mathbf{h}_i \parallel \mathbf{W}\mathbf{h}_k]))} \quad (3.8)$$

Here, $\mathbf{W}$ is the shared linear transformation matrix applied to each node’s features, and $\mathbf{a}$ is the learnable attention weight vector. The symbol $\parallel$ denotes concatenation, and $\mathcal{N}_i$ is the set of neighbors for node i . This formulation allows the model to learn which users (nodes) are more likely to influence the infection state of others, effectively learning transmission potential without relying on manually engineered features. In EpiTrack, GAT plays a pivotal role in modeling infection spread over dynamic user contact networks, learning latent spatial-temporal correlations that are otherwise hard to model through structured features. This approach is particularly effective in environments where user behaviors, contact patterns, and disease propagation are irregular and unpredictable.

Positional Transformer

Transformers utilize self-attention mechanisms to process sequential data without recurrent connections, allowing for efficient parallelization. Positional Transformers extend this concept by incorporating positional encoding to retain sequential order, making them ideal for modeling long-range dependencies in mobility and spatial-temporal forecasting. Transformers, originally developed for NLP tasks, have been increasingly applied to mobility prediction and sequential pattern recognition. Positional Transformers integrate spatial and temporal positional encoding, making them highly effective for modeling how a disease will spread across different geographic regions over time. Identifying locations with high transmission risk based on historical movement patterns. It can capture intricate relationships between infection clusters over time, even when there are irregular time

TABLE 3.4. Simulation parameters for mobility and infection spread modeling.

Parameter	Value
Total number of users	500
Simulation duration	3 hours
Initial number of infected users (t_0)	10
Distance threshold	2 meters
Intercontact time (ICT)	30 seconds
BonnMotion pedestrian speed	1.8 km/hr – 2.16 km/hr
BonnMotion mobility model	MSLAW, Random Street Mobility Model
SUMO pedestrian speed	Max 5 km/hr (Avg 3.5 km/hr)
SUMO mobility model	Random Trip Mobility Model

gaps in recorded movements. Transformers process sequential inputs via self-attention rather than recurrence. To preserve the order of sequence elements, fixed sinusoidal positional encodings are added to the input embeddings. These are computed as:

$$PE_{\text{pos},2i} = \sin\left(\frac{\text{pos}}{10000^{\frac{2i}{d}}}\right), \quad PE_{\text{pos},2i+1} = \cos\left(\frac{\text{pos}}{10000^{\frac{2i}{d}}}\right) \quad (3.9)$$

Here, pos is the position index in the sequence, i is the dimension index, and d is the model dimension. This enables the model to handle long-range dependencies and irregular timestamped sequences efficiently. In EpiTrack, Positional Transformers are used to forecast regional spread trends across spatial-temporal dimensions, learning from large-scale movement data.

3.6 Results and Discussion

To evaluate the efficacy of the proposed EpiTrack framework, we developed a synthetic simulation environment using realistic mobility traces generated from two platforms: SUMO (Simulation of Urban Mobility) and BonnMotion. Table 3.4 summarizes the core parameters used to define the epidemic and mobility scenarios. These include simulation duration, pedestrian speed, and mobility models for both platforms.

SUMO-based mobility simulation was implemented using real-world map data from OpenStreetMap (OSM). A specific region of Los Angeles was selected by specifying latitude

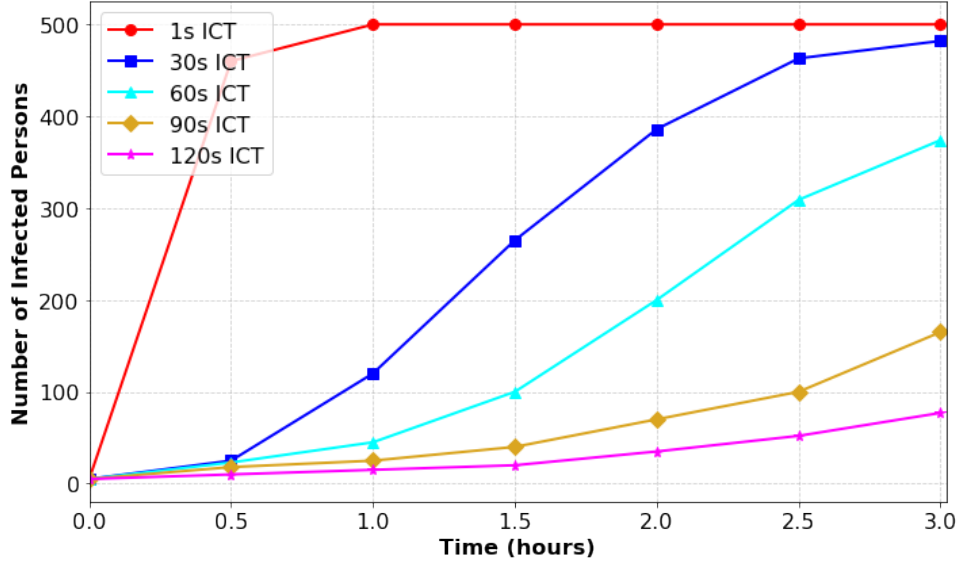


Fig. 3.6: Impact of varying inter-contact time (ICT) in the spread of the infection.

and longitude coordinates and exporting it as a .osm file. In the command window, this .osm file was combined with an .xml file containing details for roads, sidewalks, junctions, vehicular and pedestrian paths, and traffic configurations. Using randomTrips.py, we then generated random pedestrian trajectories and compiled a .rou.xml file describing route distributions, speeds, and movement edges. A configuration file comprising the .net, .rou.xml, and poly.map files was created and executed via the SUMO GUI, producing simulation output in .xml format. This file includes key parameters such as traffic speed, node location, and pedestrian trajectories. These mobility traces are subsequently preprocessed and used to model infection spread under varying contact conditions. BonnMotion similarly supports map-based mobility trace generation using OpenStreetMap and the OpenRouteService (ORS) interface. ORS enables the creation of realistic path-based simulations by defining mobility routes using an XML-based schema. BonnMotion allows users to export .map files from OSM and tailor them using inbuilt customization functions. By modifying the source code, one can adjust node locations, movement speed, route types, and other parameters. This enables simulation of diverse mobility patterns, including pedestrian or vehicular motion, reflecting realistic movement behavior. Such traces are vital for analyzing contact-based transmission dynamics and for conducting large-scale epidemic simulations in urban environments.

To demonstrate the efficacy of the proposed mobility traces monitoring and outbreak prediction proposed model and how the inter contact time have an impact on the trend of infected population, we present Figure 3.6. In our simulations, we assumed that infected individuals were actively symptomatic (e.g., coughing) while walking, and inter-contact duration determined infection transmission. Intercontact time takes account the duration of time an uninfected person spend with the infected person. We take the assumption that if the disease is very contagious there are chances that it will spread instantly even if an infected person spend merely a second with a healthy person, whereas if the uninfected person have to spend more time with infected person inorder to have a carrier of infection chances of getting the disease would be, the reason is that while walking there are reduced chances of random people be in point of close contact for longer duration, which is the scenario we have considered. Other factors are also needed to be taken into account such as if the infected person is wearing a mask or if they are coughing and in the open air environment or traveling in bus/subway etc. In our simulation settings we have considered that the infected person is coughing and has the contact with uninfected people for different duration of time while walking. We have considered the inter-contact time ranging from 1 second to upto 2 minute with an interval of 30 seconds and run the simulations. With the varying mobility and selecting random infected person distribution in the start of each simulation, when we run the simulations for 3 hours the spread of infection is exponential when ICT is 1 second, i.e., the healthy person would immediately receive the virus/infection while being in close proximity of person under the influence. From the figure it can be seen that within the duration of 30 minutes approximately 90% of the healthy population is infected from the carrier population short duration of the intercontact time. The spread rate has significantly reduced with a slight increase of intercontact time, i.e., 30 seconds. And for the longer time of contact with infected person about a duration of 2 minutes the the rate of spread has significantly reduced to 18% even after 3 hours. Hence, it can be infer that if the virus can pass on to the people quickly that it has a faster spreading nature compared to the one that takes longer to spread and the number of infected people would also vary.

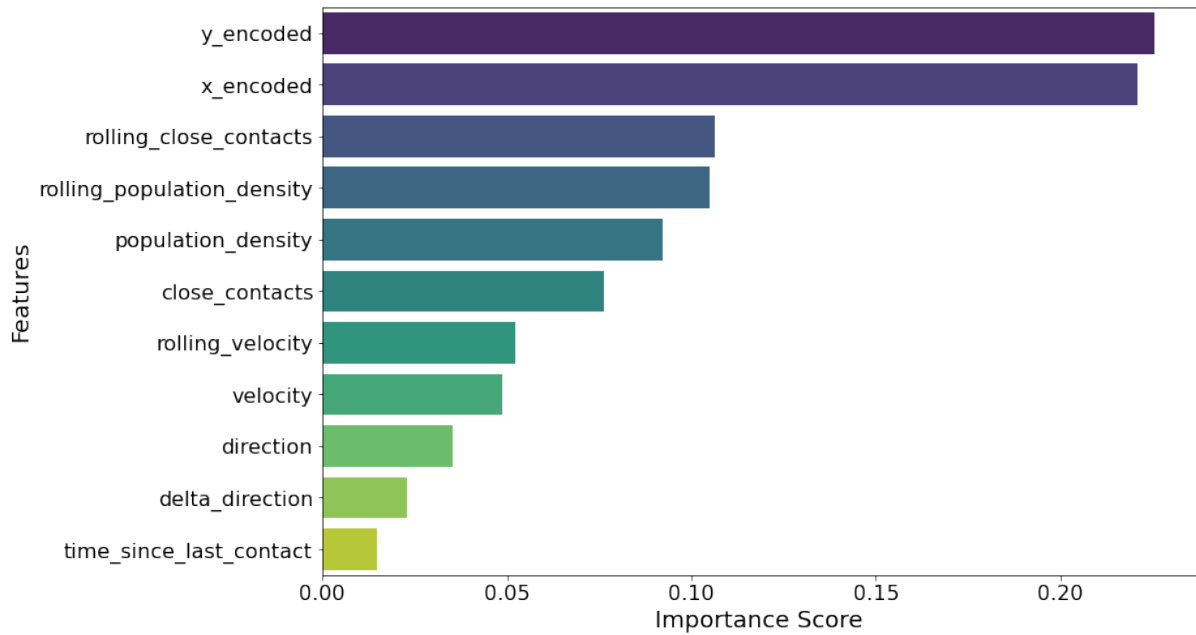


Fig. 3.7: Feature importance for infection forecasting.

TABLE 3.5. Performance comparison of ML models for infection risk prediction.

Model	Accuracy	AUC	Precision	Recall	F1-Score	Specificity
Random Forest	0.783	0.815	0.824	0.787	0.813	0.772
LSTM	0.763	0.831	0.862	0.717	0.785	0.845
BiLSTM	0.803	0.856	0.849	0.791	0.828	0.862

As illustrated in Figure 3.7, spatial features such as x encoded and y encoded dominate, reflecting their role in localizing user activity. Population density and close contacts also rank high, validating their relevance in predicting risk hotspots. Rolling features (e.g., rolling velocity, rolling close contacts) capture temporal trends by computing moving averages or variances over time, smoothing out noise and providing richer signals for downstream models. These are particularly valuable in time-series applications where behavior evolves gradually. SHAP, Gini importance, and permutation analysis were used to validate feature contributions. Results suggest that spatial location, user proximity, and contact frequency are strong predictors of exposure risk.

Based on the feature-based modeling results presented in Table 3.5, the BiLSTM model achieves the strongest overall performance for infection risk prediction, with the highest

accuracy (0.803), AUC (0.856), and F1-Score (0.828), highlighting its ability to effectively capture temporal dynamics in user mobility and exposure history. While LSTM achieves the highest precision (0.862) and specificity (0.845), it sacrifices recall, suggesting a more conservative detection approach. On the other hand, Random Forest delivers stable results across most metrics and offers strong interpretability, making it a solid baseline. The feature importance analysis further supports this performance, with spatial encodings (x/y), rolling contact metrics, and population density emerging as key predictors. These engineered features capture spatial and temporal patterns more effectively than raw input, validating the benefit of combining both statistical and deep learning models in EpiTrack’s architecture. The Figure 3.8 illustrates the evaluation metrics of two deep learning architectures applied to graph-based mobility data. The GAT model achieves significantly higher recall (near 0.98), indicating its strong ability to capture potential infection cases, which is critical for outbreak prevention. However, it shows relatively lower specificity, suggesting it may overestimate risk by flagging more false positives. On the other hand, the Positional Transformer provides a more balanced performance across all metrics. It incorporates both spatial connectivity and temporal information, leading to better generalization in time-sensitive transmission modeling. Its improved specificity and competitive precision demonstrate the model’s capability to detect high-risk cases while minimizing unnecessary alerts. These results highlight the trade-off between sensitivity (recall) and precision in predictive modeling. GAT excels at early detection by aggressively flagging possible exposures, while Positional Transformers ensure more stable predictions by leveraging timing cues in dynamic contact graphs. Together, they offer complementary strategies for real-time infection monitoring in EpiTrack.

Figure 3.9 illustrates the effectiveness of the LSTM-based model in forecasting human mobility trends, which is essential for identifying potential high-risk zones in epidemic spread modeling. In Figure 3.9a, the distribution of prediction errors is shown. The histogram reveals that the majority of prediction errors are centered tightly around zero, indicating that the LSTM model maintains high precision and stability in predicting movement patterns. This low variance in prediction error reinforces the model’s ability

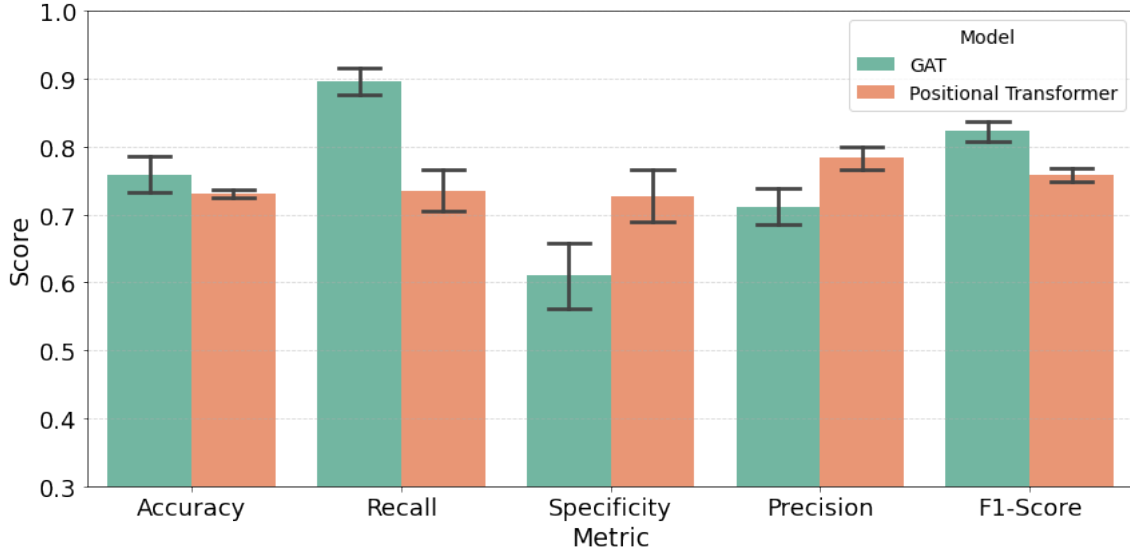


Fig. 3.8: Comparative performance of GAT and Positional transformer models for infection risk prediction across key metrics.

to generalize across different trajectory inputs. In Figure 3.9b, the actual vs. predicted mobility trends are plotted over time. The solid blue line represents the actual movement trend derived from observed user trajectories, while the dotted orange line denotes the LSTM-predicted movement trend. The shaded area represents the confidence interval, highlighting the range of expected deviations around the predicted values. As evident from the figure, the predicted movements closely follow the actual trend across time steps, capturing both periodic and transitional dynamics. This demonstrates the LSTM’s strength in modeling long-term dependencies in sequential mobility data. Such accurate movement forecasting is critical in the context of EpiTrack, as it enables early identification of movement hotspots, supports exposure prediction, and allows proactive decision-making for epidemic containment.

Figure 3.10 shows the epidemic invasion tree with the left panel illustrates the uncontrolled spread of infection when no SmartApp guidance is used, with a dense network of red trajectories indicating infected user paths. In contrast, the right panel shows a significant reduction in infection transmission under SmartApp compliance, as reflected by the prevalence of green (healthy) trajectories and limited red paths. Each line represents a user’s movement from origin to destination, where color denotes infection status. This side-by-side invasion tree visualization highlights the effectiveness of EpiTrack in curbing

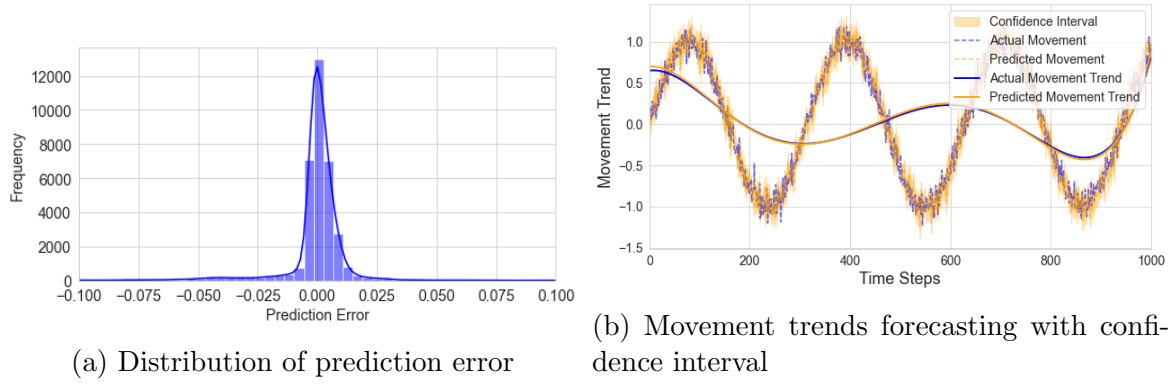


Fig. 3.9: Movement trends using LSTM

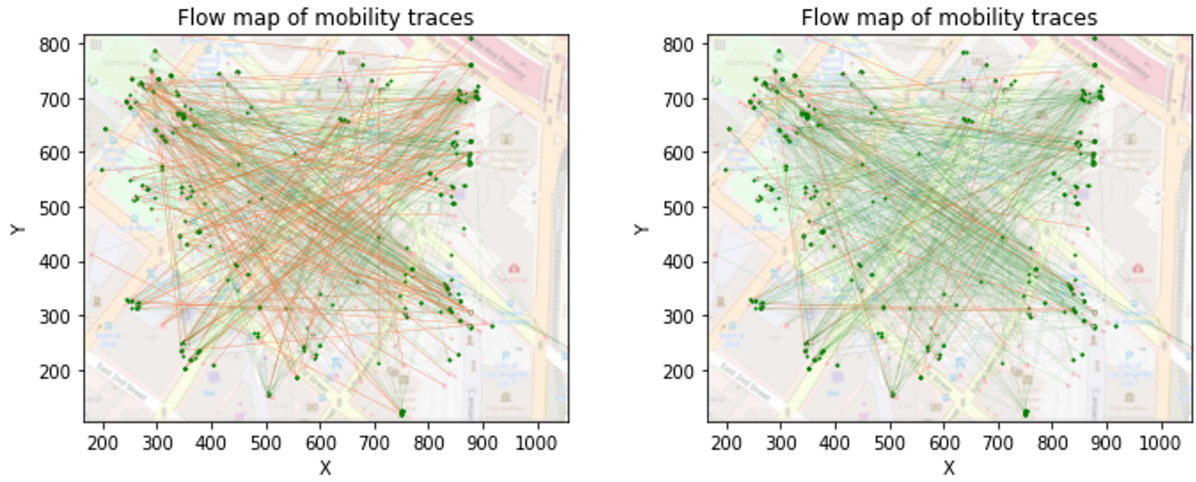


Fig. 3.10: Impact of EpiTrack compliance on epidemic spread through mobility traces.

outbreaks by influencing user behavior through timely alerts, thereby limiting the reach and intensity of disease propagation.

3.7 Conclusion

This chapter introduced an AI-powered epidemic prediction framework that utilizes smartphone collected mobility data to model and forecast the spread of infectious diseases. By capturing dynamic patterns such as user trajectories, inter-contact durations, and spatial density, the system enables accurate estimation of infection timing and identification of high-risk zones. The integration of diverse AI models—including Random Forest (RF), Long Short-Term Memory (LSTM), BiLSTM, Graph Attention Networks (GAT), and Positional Transformers allows the framework to learn from both explicit features and latent

interaction graphs, addressing spatial, temporal, and relational complexities in outbreak modeling. The use of real-time mobility traces from smart devices facilitates proactive outbreak monitoring, timely risk assessment, and optimized intervention planning. This data-driven approach empowers public health authorities to implement targeted measures, such as localized lockdowns, social distancing protocols, and resource prioritization, based on predictive insights. The study further underscores the importance of SmartApp compliance; higher participation in digital contact tracing significantly limits the spread of infection. Ultimately, by combining non-intrusive mobility sensing with intelligent predictive modeling, EpiTrack offers a scalable, adaptable solution for mitigating future epidemics and reducing their socio-economic impact.

CHAPTER 4

Proactive Reliability Modeling to Enable Resilient 6G Systems

4.1 Introduction

4.1.1 *Motivation*

The next generation of wireless networks, particularly 6G, is envisioned to support mission-critical applications that cannot tolerate service disruptions—such as remote healthcare, autonomous vehicles, smart mobility systems, UAV control, and V2X communication. These use cases demand ultra-reliable, zero-touch connectivity with strict guarantees on latency, availability, and continuity. The ubiquitous 5G and beyond (5GB) wireless networks are envisioned to achieve massive capacity (upwards of 1000 times) and connectivity (billions of connected users) [175, 176], mmWave [177], and network virtualization [178], each with a diverse set of performance and service requirements [179].

However, despite their potential, modern cellular networks continue to face numerous challenges including scalability, reliability [180], traffic management, latency, spectrum availability [181], load balancing [182], and security [183]. A particularly critical concern is the increased complexity and densification of wireless infrastructures. From 2G to 5G, the number of configuration parameters per site has increased from 500 to over 2000 Figure 4.1, with a similar growth in the number of cell sites per region [184, 185]. Such complexity renders network performance increasingly fragile [186], with system-level reliability still falling short—averaging below 98.78% in 5G [187]—despite the need for “six-nines” reliability (i.e., $>99.9999\%$) in emerging use cases such as remote surgery, autonomous vehicles, UAVs, smart grids, and mission-critical V2X applications [188].

To improve operational resilience, the Self-Organizing Network (SON) and Network Automation Functions (NAF) paradigms were introduced, standardizing automation in configuration, optimization, and self-healing via 3GPP Release 10 and beyond [189, 190].

Core SON/NAF functions include Inter-Cell Interference Coordination (ICIC), Coverage and Capacity Optimization (CCO), Mobility Load Balancing (MLB), and Mobility Robustness Optimization (MRO) [191, 192]. These functionalities offer self-adaptation, cost-efficiency, and intelligent network management. However, they remain predominantly reactive in nature—detecting anomalies or performance degradation after service quality has already been compromised.

This reactive design falls short of meeting the stringent demands of ultra-reliable, intelligent 6G systems. Traditional fault detection in cellular networks still heavily relies on manual inspection or offline logs such as MDT reports, making them ill-suited for real-time anomaly mitigation or proactive maintenance. As highlighted in [193–195], simultaneous operation of multiple SON functions often leads to logical, resource, policy, and parameter conflicts, further exacerbating service degradation. Additionally, software bugs, misconfigured intents, and human errors continue to trigger failures in high-density and high-mobility environments [196, 197]. While real-time monitoring using telemetry data such as MDT (Minimization of Drive Tests) logs may assist in responsiveness, proactive reliability assurance cannot be achieved by live data alone—especially under the expectations of six-nines reliability. In fact, reliance on reactive or real-time-only monitoring becomes increasingly insufficient as networks scale. As shown in prior studies [185], the probability of outages rises exponentially with increasing parameters and network complexity. While real-time monitoring using telemetry data such as MDT (Minimization of Drive Tests) logs may assist in responsiveness, proactive reliability assurance cannot be achieved by live data alone—especially under the expectations of six-nines reliability. In fact, reliance on reactive or real-time-only monitoring becomes increasingly insufficient as networks scale. As shown in prior studies [185], the probability of outages rises exponentially with increasing parameters and network complexity. This shortcoming is especially critical for mission-driven applications such as smart healthcare, real-time mobility monitoring, and autonomous transportation systems, where even brief service interruptions can lead to operational failures, safety risks, or loss of life. These domains demand predictive, self-healing capabilities to maintain uninterrupted connectivity and

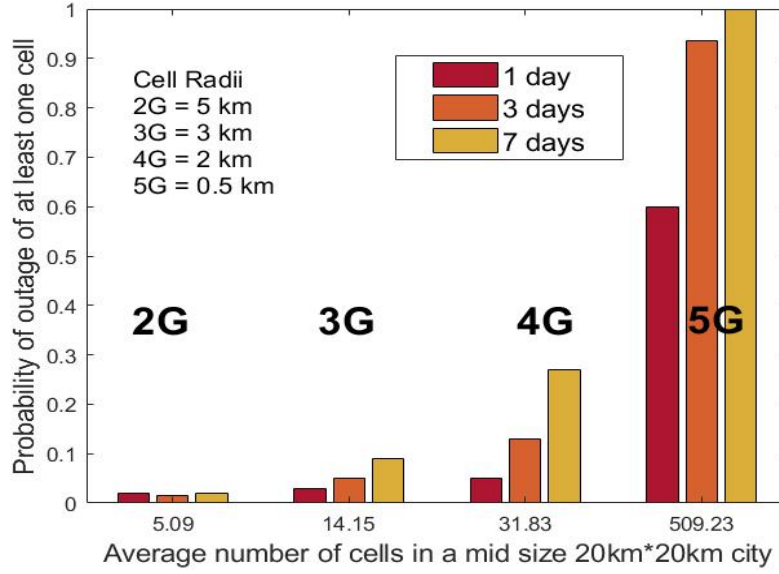


Fig. 4.1: Misconfiguration probability of a single parameter as the number of configurable settings increases.

dependable decision-making.

To address these fundamental roadblocks, we propose a predictive anomaly management framework that mathematically models network behavior using stochastic methods. This framework transitions from reactive to proactive reliability management, using Discrete-Time Markov Chains (DTMCs) to model system state transitions and identify early-stage indicators of degradation. By isolating sensitive parameters and analyzing their role in triggering failures, this framework enables preemptive interventions—enhancing network resilience and ensuring optimized resource allocation in future 5G and 6G deployments.

4.1.2 Related Work

A conflict/fault free SON/network automation function (NAF) orchestration is inevitable to enable the reliable network operations. Thus, conflict avoidance in NAFs is a crucial yet an understudied research problem. An earlier study [195] has identified a taxonomy to classify different conflicts such as KPI conflicts, parameter conflicts, logical dependency conflicts, and measurement-based conflicts. The authors also presented several self-coordination mechanisms for the conflict resolution. In literature, the most extensively studied framework to avoid conflicts is the co-design conflict coordination algo-

rithm [194, 198–201]. Several studies have developed NAF/SON conflict coordination schemes, in particular for the MRO and MLB SON functions. The authors in [195] suggested Trigger-Condition-Action (TCA) policies for the conflict coordination of the MLB and MRO functions. Another SON COordinator (SONCO) framework for the conflict resolution of the two distributing NAF functions (MLB and MRO) is proposed in [202]. The reinforcement learning tunes the arbitrations, i.e., Cell Individual Offset (CIO) parameter for the MLB and handover hysteresis parameter for the MRO function on the basis of the operator assigned priorities. For the assurance of conflict free operation of CCO and load balancing functions, an optimization problem is formulated to optimize the antenna tilt, transmit power, and CIO as a soft parameter in [199]. An operator’s objective-driven SON coordination scheme is devised in [203] for three SON functions namely CCO, MLB, and MRO. The authors devised two coordination schemes, i.e., policy-based and objective-driven coordination. In the policy-based conflict resolution, CCO is being prioritized over MRO and MLB SON functions. Whereas, the objective-driven coordination determines the subgroup of the requests that is conflict-free that can prominently improves the utility and can select the concurrent execution of the two SON functions. Thus, based on the multiattribute utility theory-based objectives, it is possible to achieve subgroup of the requests that is conflict-free with the help of optimization problem. Most of the proposed approaches in the literature are considering conflict coordination specific to only two SON functions [194, 195, 198, 199, 201, 202]. In [204] the authors proposed the enhancement of SON functions in millimeter wave bands. The core idea is that the users report the additional reflection environment map, that renders a fast recovery mechanism from the loss of connectivity caused by the poor channel conditions.

Despite the reviewed advancements, the state-of-the-art *modus operandi* is reactive, i.e., the fault detection and compensation mechanisms for automation conflicts are activated once the fault/outage has already been occurred. Hence, even if misconfiguration/fault is detected, a disruption in service in the order of several seconds to minutes is inevitable. Therefore, without transforming the way conflicts are managed in the emerging networks, achieving high reliability may remain an elusive goal in 5G&B networks [205]. In this

TABLE 4.1. Comparison of state of the art literature and our work.

Ref No.	SON function	Parameters	Contributions
Lateef et al. 2015 [195]	CCO, ICIC, COC, MLB, etc.	Antenna tilt, Tx power, CIO, TTT, hysteresis, filter co-efficient	Categorization of possible conflicts among the distinctive SON functions
Rojas et al. 2020 [198]	MLB, MRO	CIO, TTT, hysteresis	A machine learning framework for self adaptation of SON function and network optimization once conflicts happened
Asghar et al. 2018 [199]	CCO, LB	Tx power, antenna tilt	Simultaneous coverage, capacity, and load balancing optimization of SON functions with the help of soft and hard cell parameters.
Moysen et al. 2018 [200]	MLB, MRO	CIO, hysteresis, and TTT	A ML model and multi-objective algorithm that resolves SON conflicts and predict network performance based on historical UE measurement
Bag et al. 2020 [201]	ICIC, CCO	Power factor, antenna tilt, edge to center boundary	Recommender System that determine coverage and capacity in different cell and recommend configuration parameters to maximize the multi-objective performance of both SON functions
Iacoboaiea et al. 2014 [202]	MRO, MLB	CIO, hysteresis	Reinforcement learning framework SONCO to resolve conflicts by jointly optimizing both SON functions based on the operator priorities
Frenzel et al. 2015 [203]	CCO, MLB, MRO	CQI, CIO	The conflict resolution of objective-driven SON functions by giving CCO SON function highest priority compared to the MLB and MRO. The SON objective manager concept is modeled as a constraint optimization problem
Our work	Combination of any 4 NAF functions (MLB, CCO, ICIC, EE, MRO, etc.)	Neighbor list, Radio bearer assignment, Downlink Tx power, antenna tilt and azimuth, switching on/off cell, handover parameters (CIO, hysteresis, and TTT), MIMO configuration	Discrete Time Markov Chain (DTMC) analysis is performed using four of the most impactful network parameters in terms of coverage reliability, thus joint optimization of 4 NAF functions

chapter, we propose a generalized fault prediction framework that can enable this transformation in the way automation caused or otherwise faults are handled in the network. The proposed model, unlike the state-of-the-art is not limited to the structural aspects of the cells but takes into account the impact of parameter misconfiguration for the network fault scenarios. Our proposed framework not only characterizes the reliability of the automation enabled networks, but it can also predict future network behavior in terms of normal or suboptimal (partial/complete outage) performance to improve the network reliability. A network equipped with state of the art automated but still reactive fault detection solution can offer more reliability than a network with manual operation by reducing the potential outages time from hours to minutes. However, with state-of-the-art reactive fault detection or post-conflict reliability enforcement, network cannot offer Ultra Reliable Low Latency Communications (URLLC), a major and yet to be realized 5G and beyond use case. In contrast, the proposed framework by predicting the faults or automation conflicts before they happen can enable preemptive maintenance, thereby reducing the cell outage time to effectively zero. Thus, proposed solution offers the novel proactive approach that can act as key enabler the URLLC when it comes to fault detection or ensuring network reliability.

A detailed comparison of our proposed solution with the relevant solutions in the state-of-the-art literature is given in Table 4.1. More specifically, the presented model can predict the expected time and cause (parameter misconfiguration) of the suboptimal behavior by considering the expected conflicts among primary parameters. No existing solution offers such capability as explained above. Such capabilities can be leveraged for detection and isolation of early developing faults in the mobile networks. For the prediction of the future trends in wireless networks such as traffic, and fault prediction, Machine Learning (ML) is being leveraged in the recent years [196,198,200,201,206,207]. Fault prediction using ML has two possible approaches; 1) using historic network fault logs to create a regression or Deep Learning (DL) model [198,200,207], 2) creating a state space Markov chain model and predicting the next states. For the purpose of prediction of future faults, first approach requires large amount of historical data. In contrast,

the second approach can work with small amount of historical data, as the data is only needed for initializing the transition probabilities. However, it does require more domain knowledge aware mathematical modeling of network parameters and behavior. Benefiting from the domain knowledge, our proposed proactive fault prediction framework leverages the later approach. The additional benefits of proposed solution include need far less amount of historical data needed to create the model, low computational complexity, and ability to incorporate domain knowledge that in turn can capture in live network behavior that deep learning based black box models for fault prediction may miss. In [198] an ML based zero-touch coordination framework is proposed to automatically learn the dynamics based on the history between the selected SON functions (MRO and MLB) for the network optimization. In [200] ML is used for the network performance prediction of the SON functions (MLB, MRO) based on the network history. The authors concluded that there is a trade-off related to SON function performance. An ML based zero-touch coordination framework to automatically learn the dynamics based on history between the selected SON functions (MRO and MLB) that assists the network optimization task is proposed in [198]. In this chapter, we propose a fault/misconfiguration prediction framework by using geometrical distribution for DTMC that can detect the upcoming network performance degradation. The key insights gained from the presented analysis can be exploited to either design the predictive self-healing functions or preemptively scheduling the parameter configuration and status checking routines to avoid the degraded performance altogether. The proposed outage/fault prediction model for the conflict avoidance; thus takes a step towards the reliable 5G&B networks.

4.1.3 Contributions

The contributions of this chapter can be summarized as follows:

1. We propose a realistic probabilistic model that quantifies the likelihood of parameter misconfigurations due to manual errors or conflicting network automation functions. This model supports early detection of suboptimal configurations, enabling proac-

tive troubleshooting and reliability enhancement in dense cellular environments.

2. Using Discrete-Time Markov Chains, we analyze long-term network behavior by modeling transitions between different reliability states. The framework focuses on four critical parameters affecting coverage and demonstrates how small deviations can significantly impact user QoE.
3. We model overlapping parameter dependencies across various automation functions and identify key parameters whose misconfiguration has the highest degradation risk. This facilitates prioritization of parameter checks and supports smarter configuration scheduling.
4. To observe the impact of the multi-parametric conflicts in network automation function, we devise two performance metrics, i.e., *mean first passage time and limiting or steady state distribution*. By leveraging these metrics, we can evaluate the probability of the network degradation and perform the multi-parameter optimization.

4.1.4 Chapter Organization

The rest of the chapter is organized as follows: Section 4.2 describes the reliability probability analysis of the NAF conflicts and the DTMC model. The performance evaluation and results are presented in Section 4.3. Section 4.4 describes the utility of the proposed fault prediction model. Finally, Section 4.5 concludes the chapter.

4.2 Stochastic Modeling and Reliability Analysis

In the network automation, there are several hundreds of configuration parameters that are related to SON/NAF, yet the majority of them are rarely changed. However, among these parameters, some are considered as the crucial parameters, that belong to the multiple network automation functions and have a higher probability of being reconfigured. Some of the important NAF and their related configuration parameters are shown in

Table 4.2. In general, the categorization of a parameter as a primary or secondary parameter, can be performed based on the *network logs*, *expert domain knowledge* or *by leveraging ML based Shap analysis*. As mentioned earlier, the hidden, logical, parametric or objective-based conflicts between the network automation function parameters can increase the risk of parameter misconfiguration and can raise concerns regarding the network reliability. Therefore, we analyze the probabilistic reliability behavior of the 5G&B networks considering the conflicting SON functions and their associated parameters. The

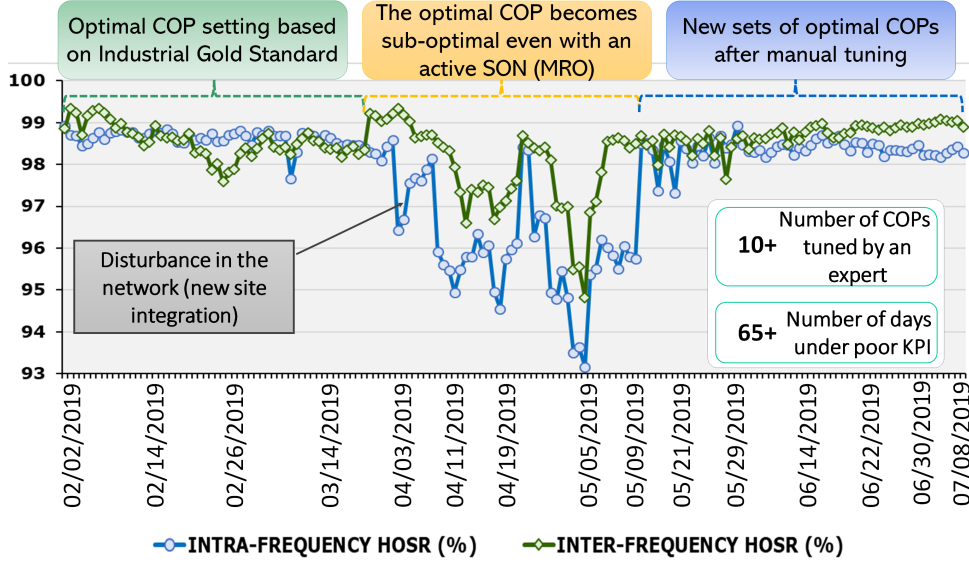


Fig. 4.2: Suboptimal configuration of the Network Automation Function parameter.

misconfiguration of the network automation function parameters can degrade the performance of the entire network, i.e., when the network undergoes suboptimal settings it requires time to detect the underlying cause, to activate the compensation actions, and consequently to stabilize the network. Figure 4.2 demonstrates a real network-based example of the misconfiguration of the MRO function. The data is collected from a mobile network operator (MNO). It can be observed from the figure that the number of successful handovers within the same frequency (intra HO) and with different operating frequency (inter HO) is quite high when the Configuration and Optimization Parameters (COPs) are optimal based on the industrial Gold Standards. However, during the start of April, there is a prominent performance degradation in the MRO SON function due to the suboptimal value of one of its parameters, i.e., very high cell individual offset (CIO)

TABLE 4.2. NAF/SON functions and the related parameters.

SON Function	Description	Primary Parameters	Secondary Parameters
ANR/ PCI	ANR builds, maintains neighbor relationships. PCI automatically configures Cell Identity of eNB	N	-
ICIC	Reduces intercell interference	R	P, A, S
Coverage hole management	Automatically detects and compensates coverage holes in timely manner	P, A	H, M
eNB insertion/ removal	Self-configures home eNBs	N, P, H	-
CCO	Optimizes eNB coverage and capacity of eNB	P, A	R, H, M
Energy saving	Optimizes eNB energy consumption of eNB	P, S, M	-
MLB	Optimizes cell reselection/handover parameters	P, A, S, H	-
MRO	Automatically sets the handover parameters	H	N
Relay/repeater management	Self-configures and optimizes relays and repeaters	R, P, S	H, M

N –Neighbor list, R –Radio bearer assignment, P –Downlink Tx power, A –Antenna tilt and azimuth, S –Switching on/off cell, H –Handover parameters, M –MIMO configuration.

value towards the neighboring cell of the serving node, that caused hindrance in performing the handovers. The problem was eventually detected and solved by the careful configuration of more than 10 COPs with the help of human intervention. This example illustrates that the misconfiguration of a parameter, can have a deteriorating impact on the network reliability. The detection of such anomalous behavior requires highly trained domain experts to manually parse through the data logs. Hence, for the timely reliability provision, there is a need to analyze the probability of misconfiguration to enable the proactive parameteric conflict avoidance solutions in the 5G cellular networks.

Assumptions

For the modeling of a proactive conflict avoidance network model, following assumptions are taken into account:

- A node is assumed to be in a suboptimal state/outage when any of its parameter is reconfigured with a suboptimal value. This assumption holds true because the misconfiguration of any primary/secondary parameter can deteriorate the network performance.
- When a SON is called (reactively or proactively), it will reconfigure only one of its associated parameters per call. This assumption is built upon the fact that the reconfiguration of one parameter at a time will refrain the network to have conflicts among the interdependent parameters of different NAF [203].

4.2.1 Model Development

For the calculation of the misconfiguration probabilities, let's suppose we have N network automation functions $\varphi_{\mathcal{N}} = \{\varphi_1, \dots, \varphi_N\}$ and M_i is the number of primary and/or secondary parameters $v_{\mathcal{M}_i} = \{v_1, \dots, v_{M_i}\}$, configurable by a NAF. At any time instant, one NAF function is called and its probability is $Pr(\varphi_i)$. The probability that v_j parameter is reconfigured when the NAF φ_i gets activated is given by the conditional probability $Pr(v_j|\varphi_i)$. Let $Pr(\mathcal{S})$ be the probability of any of the network parameter to become misconfigured/suboptimal when a network automation function is called and $Pr(\mathcal{O}) = 1 - Pr(\mathcal{S})$ be the probability of v_j becoming optimal. Thereby, the conditional probability of the v_j parameter to become suboptimal given that the network automation function φ_i is called and v_j parameter gets selected is given by; $Pr(\mathcal{S} | (\varphi_i \cap v_j))$. These dependent probabilities for the NAF conflicts can be represented by the probability tree diagram as shown in Figure 4.3.

Thus, the total probability of a network parameter to become suboptimal when the NAF is called can be given by [208]:

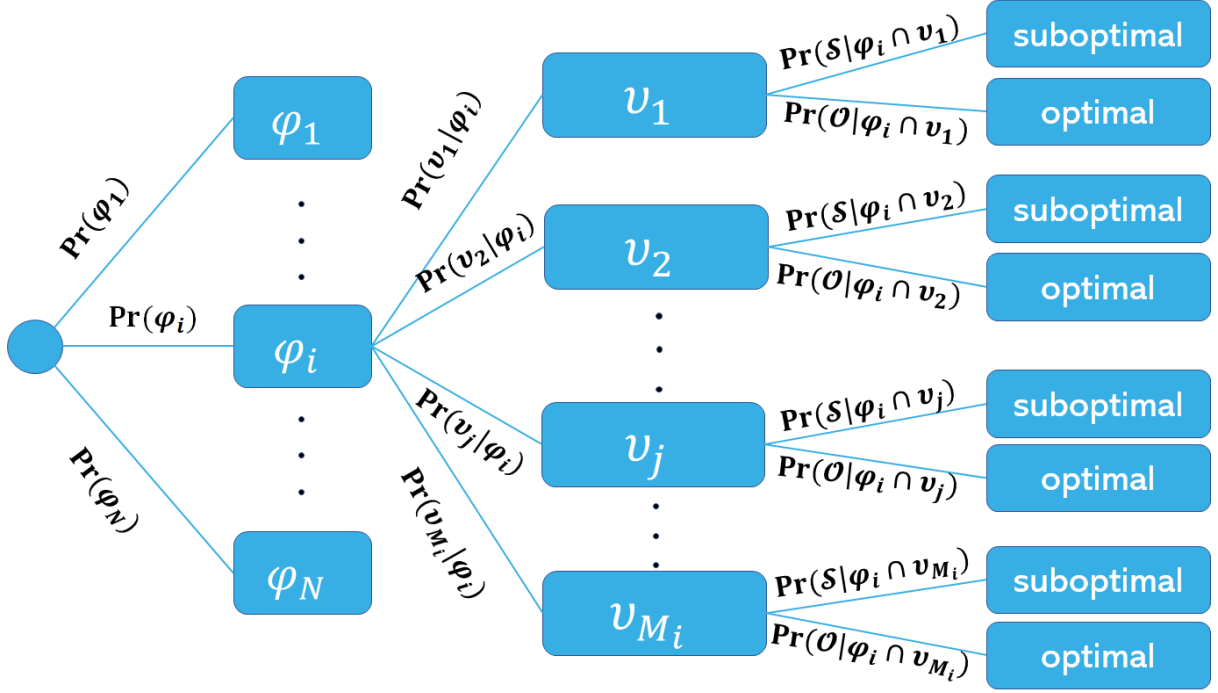


Fig. 4.3: Probabilistic tree illustrating NAF conflicts and resulting parameter state outcomes.

$$Pr(\mathcal{S}) = \sum_{i=1}^N \sum_{j=1}^{M_i} Pr(\varphi_i) \times Pr(v_j|\varphi_i) \times Pr(\mathcal{S}|\varphi_i \cap v_j) \quad (4.1)$$

Similarly, the probability of a single network parameter, for instance, the parameter handover H to become suboptimal when a NAF function gets activated is given as:

$$Pr(H_S) = \sum_{i=1}^N Pr(\varphi_i) \times Pr(H|\varphi_i) \times Pr(\mathcal{S}|\varphi_i \cap H) \quad (4.2)$$

The probability of a function to get selected is:

$$Pr(\varphi_i) = \frac{1}{N}, \text{ equally likely } \forall i = 1 \text{ to } N \quad (4.3)$$

or they can have different probabilities such that $\sum_{i=1}^N Pr(\varphi_i) = 1$. The conditional probability $Pr(v_j|\varphi_i)$ of equation (4.1) can be further classified as follows:

- If the network automation function φ_i operates on all of its M_i parameters with equal priority, then the probability of v_j given φ_i gets called is: $Pr(v_j|\varphi_i) = (\frac{1}{M_i})$.

- If the network automation function φ_i has M_{pi} and M_{si} number of primary and secondary parameters, respectively, such that $M_i = M_{pi} + M_{si}$, then the probability of v_j given φ_i is:

$$Pr(v_j|\varphi_i) = \begin{cases} \left(\frac{Pr(\text{Primary parm})}{M_{pi}} \right), & v_j \text{ is primary parm of } \varphi_i \\ \left(\frac{Pr(\text{Secondary parm})}{M_{si}} \right), & v_j \text{ is secondary parm of } \varphi_i \end{cases} \quad (4.4)$$

Equation (4.1) is used when the network automation function is activated once. However, with the passage of time, as the network automation functions are activated repeatedly, we have the Bernoulli trials with two outcomes to model this scenario: suboptimal or optimal configuration of the parameters. Based on the Bernoulli trials assumptions, i.e., the trials are independent, each trial has one of the two possible outcomes, and the probability of the outcome remains constant from trial-to-trial [208]. We are considering that the NAF are called different times of the day and are independent. If the network automation function are activated n number of times, in that case the probability of getting r successes is given as, $\binom{n}{r} p^r (1-p)^{n-r}$.

As the time progresses and the network automation functions get activated r number of times, the probability of having the optimal parameters in all activations is given by $(1 - Pr(\mathcal{S}))^r$. Therefore, the probability of getting at least one suboptimal parameter in r number of runs will be: $Pr(\text{at least one subopt parm}) = 1 - (1 - Pr(\mathcal{S}))^r$ where $Pr(\mathcal{S})$ is calculated from equation (4.1). The probabilities can be extracted from the real networks e.g., Operations and Maintenance Centre (OMC) network logs that will have information on the frequency of the network automation functions activation.

4.2.2 Markov Analysis

DTMC can be used to model the long term network behavior. It is a discrete time stochastic process on the state space S , where the conditional probability of any future event is only dependent upon the current state and is independent of the past states.

Mathematically for all y and z in S :

$$P(X_{t+1} = z | X_t = y, X_{t-1}, \dots, X_0) = P(X_{t+1} = z | X_t = y) \quad (4.5)$$

As in our network model, the transition conditional probabilities are not dependent upon time, therefore, the DTMC will be time homogeneous, and defined as:

$$P(X_{t+1} = z | X_t = y) = P(X_t = z | X_t = y) = p_{yz} \quad (4.6)$$

where p_{yz} is a one step transition probability from state y at time t to state z at time $t + 1$. For a Markov Chain with L number of states, the transition matrix $\mathbf{T}$ can be expressed as an $L \times L$ matrix:

$$\mathbf{T} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \dots & p_{1L} \\ p_{21} & p_{22} & p_{23} & \dots & p_{2L} \\ p_{31} & p_{32} & p_{33} & \dots & p_{3L} \\ \dots & \dots & \dots & \dots & \dots \\ p_{L1} & p_{L2} & p_{L3} & \dots & p_{LL} \end{bmatrix} \quad (4.7)$$

4.2.3 *Transition Matrix Properties*

For the purpose of our analysis, we have chosen four network parameters “ P – Downlink Tx Power, A – Antenna Tilt and Azimuth, H – Handover Parameters, and M - MIMO Configuration”, that intuitively have the maximum impact on the coverage reliability. A small deviation in their values can have a significant impact on the user experience [205]. The discrete time t corresponds to the activation of the NAF for the t^{th} time. At any time instant, when a network automation function is called, any of these parameters can be in one of the two states, i.e., optimal or suboptimal. The following properties govern the transition matrix $\mathbf{T}$ and are used to set the necessary conditions aligned with the realistic assumptions:

TABLE 4.3. Transition probabilities of NAF functions resulting in optimal or suboptimal network parameter configurations.

p	NAF function changes power to optimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(P \varphi_i) \times Pr(\mathcal{O} \varphi_i \cap P)$
a	NAF function changes antenna parameters to optimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(A \varphi_i) \times Pr(\mathcal{O} \varphi_i \cap A)$
h	NAF function changes handover parameters to optimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(H \varphi_i) \times Pr(\mathcal{O} \varphi_i \cap H)$
m	NAF function changes MIMO configuration to optimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(M \varphi_i) \times Pr(\mathcal{O} \varphi_i \cap M)$
p'	NAF function changes power to suboptimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(P \varphi_i) \times Pr(\mathcal{S} \varphi_i \cap P)$
a'	NAF function changes antenna parameters to suboptimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(A \varphi_i) \times Pr(\mathcal{S} \varphi_i \cap A)$
h'	NAF function changes handover parameters to suboptimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(H \varphi_i) \times Pr(\mathcal{S} \varphi_i \cap H)$
m'	NAF function changes MIMO configuration to suboptimal value	$\sum_{i=1}^N Pr(\varphi_i) \times Pr(M \varphi_i) \times Pr(\mathcal{S} \varphi_i \cap M)$

- As the number of parameters are finite, i.e., 4 in our case, therefore, the DTMC State Space S will be finite consisting of $2^4 = 16$ states.

$$S = [S_1 \ S_2 \ S_3 \ ... \ S_l], \quad l = 16 \quad (4.8)$$

- p_{yz} is positive real number between 0 and 1.

$$0 \leq p_{yz} \leq 1, \quad \forall y, z \in [1, L]. \quad (4.9)$$

- T will be a right stochastic matrix, i.e,

$$\sum_{z=1}^L p_{yz} = 1, \quad 1 \leq y \leq L \quad (4.10)$$

Transition matrix initialization

It is assumed that whenever any network automation function is activated, it operates on only one of its associated parameters per transition, this assumption is also supported by [209]. The transition probabilities are calculated as per Table 4.3. Therefore, those states cannot be reached directly (zero transition probability) in which more than one parameter is changed.

There will be a total of 16 DTMC States as defined in Table 4.4, wherein $\checkmark$ indicates that the parameter is configured with the optimal value while $\times$ represents the misconfigured value. Using the information in Table 4.2 and Table 4.3, the Transition matrix $\mathbf{T}$ is given as:

$$\mathbf{T} = \begin{bmatrix} \psi & m' & h' & 0 & a' & 0 & 0 & 0 & p' & 0 & 0 & 0 & 0 & 0 & 0 \\ m' & \psi & 0 & h' & 0 & a' & 0 & 0 & 0 & p' & 0 & 0 & 0 & 0 & 0 \\ h & 0 & \psi & m' & 0 & 0 & a' & 0 & 0 & 0 & p' & 0 & 0 & 0 & 0 \\ 0 & h & m & \psi & 0 & 0 & 0 & a' & 0 & 0 & 0 & p' & 0 & 0 & 0 \\ a & 0 & 0 & 0 & \psi & m' & h' & 0 & 0 & 0 & 0 & 0 & p' & 0 & 0 \\ 0 & a & 0 & 0 & m & \psi & 0 & h' & 0 & 0 & 0 & 0 & 0 & p' & 0 \\ 0 & 0 & a & 0 & h & 0 & \psi & m' & 0 & 0 & 0 & 0 & 0 & 0 & p' \\ 0 & 0 & 0 & a & 0 & h & m & \psi & 0 & 0 & 0 & 0 & 0 & 0 & p' \\ p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \psi & m' & h' & 0 & a' & 0 & 0 \\ 0 & p & 0 & 0 & 0 & 0 & 0 & 0 & m & \psi & 0 & h' & 0 & a' & 0 \\ 0 & 0 & p & 0 & 0 & 0 & 0 & 0 & h & 0 & \psi & m' & 0 & 0 & a' \\ 0 & 0 & 0 & p & 0 & 0 & 0 & 0 & 0 & h & m & \psi & 0 & 0 & a' \\ 0 & 0 & 0 & 0 & p & 0 & 0 & 0 & a & 0 & 0 & 0 & \psi & m' & h' \\ 0 & 0 & 0 & 0 & 0 & p & 0 & 0 & 0 & a & 0 & 0 & m & \psi & h' \\ 0 & 0 & 0 & 0 & 0 & 0 & p & 0 & 0 & 0 & a & 0 & h & 0 & \psi & m' \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & p & 0 & 0 & 0 & a & 0 & h & m & \psi \end{bmatrix} \quad (4.11)$$

wherein $\psi = \mathbf{1} - \sum_{y \neq z} \mathbf{a}_{yz}, \forall z, \mathbf{a}_{yz} \in \mathbf{T}$, to satisfy the right stochastic property.

4.2.4 Transition Matrix Analysis

Using the Chapman–Kolmogorov equation, the probability of the network moving from state y to state z after k periods is calculated by multiplying the transition matrix itself k times. If $\mathbf{p}_0$ is the initial probability state vector, then

TABLE 4.4. Discrete time markov chain (DTMC) states.

States	Parameters Configuration ✓: optimal : suboptimal			
	P	A	H	M
S_1	✓	✓	✓	✓
S_2	✓	✓	✓	
S_3	✓	✓		✓
S_4	✓	✓		
S_5	✓		✓	✓
S_6	✓		✓	
S_7	✓			✓
S_8	✓			
S_9		✓	✓	✓
S_{10}		✓	✓	
S_{11}		✓		✓
S_{12}		✓		
S_{13}			✓	✓
S_{14}			✓	
S_{15}				✓
S_{16}				

$$\mathbf{p}_k = \mathbf{p}_0 \mathbf{T}^k \quad (4.12)$$

where $\mathbf{p}_0 = [p_1 \ p_2 \ p_3 \ \dots \ p_L]$ and $\mathbf{p}_k = [\gamma_1 \ \gamma_2 \ \gamma_3 \ \dots \ \gamma_L]$, $\mathbf{p}_k$ is the k^{th} probability vector for the l states.

In order to find out the number of transitions after which the network first becomes suboptimal for each state of the DTMC, we define ***mean first passage time*** performance metric, i.e., expected time for the first occurrence of fault. The mean first passage time provides a measure of the network to be suboptimal due to each of the parameter's misconfiguration status connected to the network automation functions [210]. The mean first passage time from state y to state z is the expected number of transitions before the network first reaches to the state z , given that the network is currently in state y . Denoted by m_{yz} , it is evaluated as:

TABLE 4.5. Mobile network setting with initial probabilities.

Probability of the network automation function to be selected	Equally Likely, 1/9
Probability of the primary parameter to be selected	70% of the time
Probability of any secondary parameter to be selected	30% of the time
Probability of a NAF to correctly configure any of its parameter (optimal value)	90% of the time
Probability of a NAF to misconfigure any of its parameter (suboptimal value)	10% of the time

$$m_{yz} = 1 + \sum_{k \neq z} p_{yk} m_{kz} \quad (4.13)$$

The ***limiting or steady state distribution*** is a measure to indicate the fraction of time the network will be in each of the states [207]. It is defined as:

$$\pi = [\pi_1, \pi_2, \pi_3, \dots, \pi_l] \quad (4.14)$$

$$\pi_z = \lim_{n \rightarrow \infty} P(X_t = z), z \in \mathbf{S} \quad (4.15)$$

This steady state distribution can also be used to find the cost of having suboptimal behavior. The long run expected loss due to the unreliable network behavior can be calculated as:

$$\text{Revenue loss} = \sum \pi_z c(z) \forall z \in \text{Suboptimal States} \quad (4.16)$$

where $c(z)$ is the loss incurred due to the suboptimal network behavior and π is the final state distribution.

4.3 Numerical Results and Discussion

For the numerical analysis, we consider the network settings of the network automation functions and their associated parameters as described in Table 4.6. As we do not have operator provided past logs/reports of outages or network misconfigurations, we assumed the initial probabilities for the numerical analysis, that are given in Table 4.5, unless stated otherwise. The probabilities can be extracted from the real networks, for instance,

OMC network logs have the information on the frequency of the network automation functions activation. Based on the historical traces, and the parameters used by the network agents, we can deduce the actual priority with which the parameter is categorized as a primary or secondary parameter for a network automation function. Similarly, what is the probability of the actual choice of selection. However, due to the unavailability of the proprietary network data, we are considering a numerical illustration with the initial probabilities as given in Table 4.4. It can be observed from the table that the probability of correct parameter configuration associated to a network automation function is much higher compared to the parameter misconfiguration.

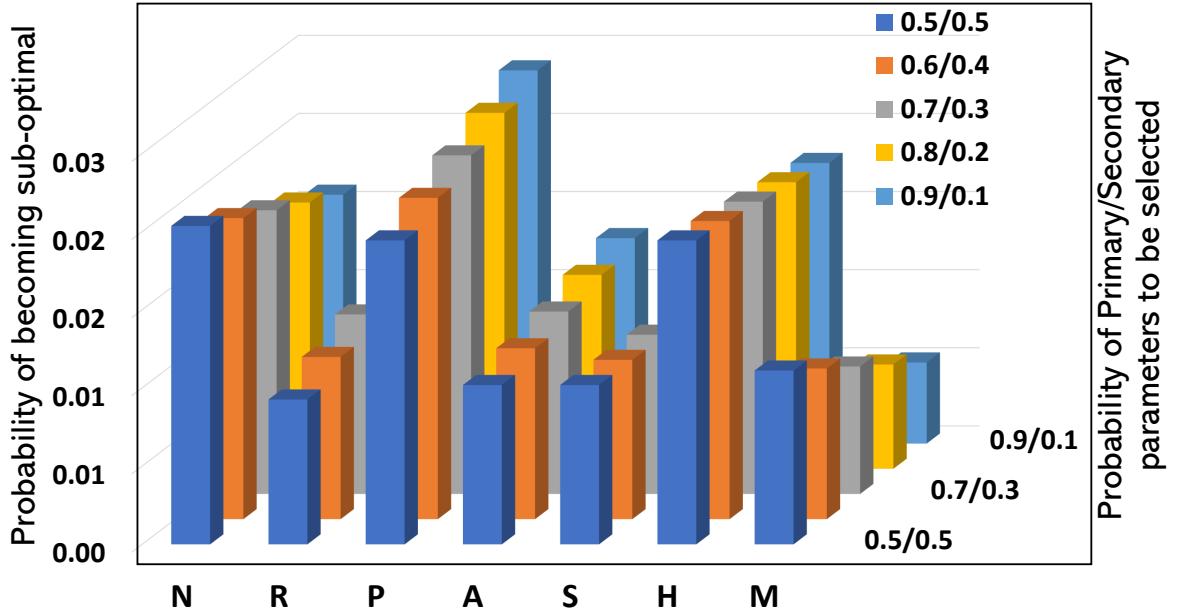


Fig. 4.4: Probability of parameters to become suboptimal.

In Figure 4.4 we plot the probability of the network parameters to become suboptimal when configured for the different network automation functions. For this figure, we leverage equation (1) with different probability of selection of the primary and secondary parameters. The result conforms to the intuitive notion that the overlapping parameters in multiple network automation functions as primary parameters have the highest probability to become suboptimal.

The power parameter P (a primary parameter for 6 network automation functions in Table I) has the highest probability of misconfiguration followed by the handover H and

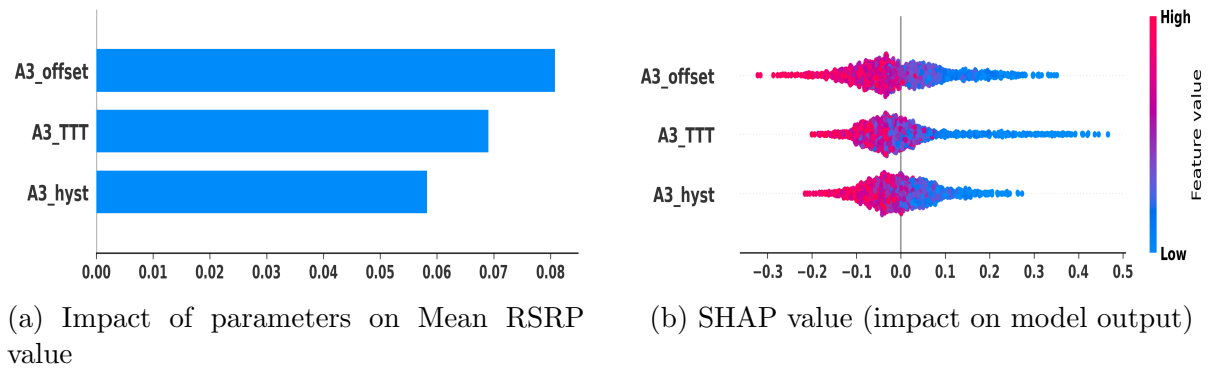


Fig. 4.5: Impact of different parameters on network performance.

TABLE 4.6. Description of the simulation parameters.

System Parameters	Value
Number of BSs	3
Number of Users	50
Operating frequency of BSs	1.7GHz
Bandwidth of BSs	15MHz
Pathloss exponent	3
Shadowing standard deviation	4dB

neighbor list N parameters. In addition, it is worth noting that there is an increasing trend in the misconfiguration probability for some parameters as the probability of primary parameter to be selected is increased from 0.5 to 0.9, while some parameters have the decreasing trends. For instance, MIMO Configuration M parameter in Table 4.2 is listed mostly as a secondary parameter and with the decreasing probability of secondary parameters to be selected along z-axis, the probability of such parameters to become suboptimal reduces. While the parameters which appear mostly as the primary parameters e.g., power P , antenna tilt and azimuth A , these parameters show increasing trends across z-axis. Thus, the parametric misconfiguration probabilities that are calculated by leveraging the proposed reliability analysis, can be beneficial to prioritize the verification of each parameter based on its estimated misconfiguration probability.

In the real wireless networks, some network automation functions are called more frequently compared to others, such as, ICIC, MLB, and MRO network automation functions compared to ANR or PCI functions. Equation (1) and (2) state that the probability of selection of a particular network automation function has a direct impact on the mis-

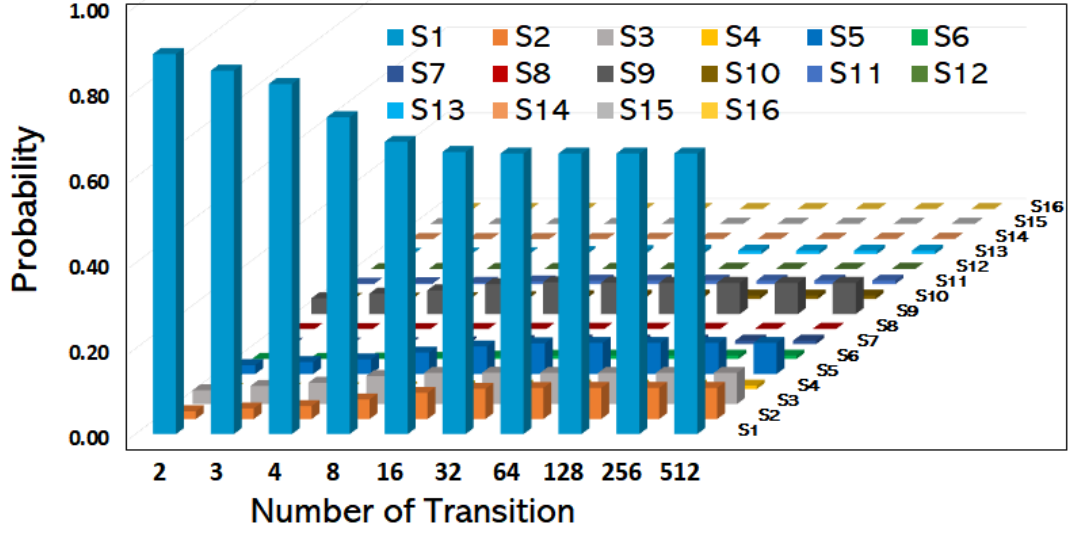


Fig. 4.6: Steady state distribution of the network.

configuration of the parameters. Therefore, these ‘risky’ NAF which have a large number of ‘sensitive’ parameters, they should be scheduled relatively less frequently. Results in Figure 4.5 corroborate this contemplation. For the figure, we leverage a 3GPP-compliant system level 5G simulator [211]. A simulation area of $1.5\text{km} \times 1.5\text{km}$ is used for the data generation. User mobility type is random way point and the speed remains constant for a user. The details of the network level simulation parameters are summarized in Table 4.5. The MRO network automation function tunes three mobility parameters related to the intra-frequency handover, i. e., A3 offset, A3 TTT, and A3 hysteresis to achieve the optimized objectives. A range of 2dB to 15dB is used for the A3 offset while for A3 TTT 7 values from 64ms to 512ms with an interval of 64 ms are used, and 1dB to 10dB range is considered for A3 hysteresis to generate the COP-KPI data. The figure illustrates that these three handover related parameters have a varying impact on the network mean Reference Signals Received Power (RSRP) values. When we perform the Shapley analysis of the handover parameters, to analyze the sensitivity of uncertain variables, it can be observed that the A3 offset has the most substantial impact on the RSRP value and thus for triggering the handover as shown in Fig 4.6 (a) and (b). Therefore, it can be categorized as a sensitive parameter. This figure is depicting the fact that for the SON functions such as MRO and MLB, where handover being the primary parameter, the configuration frequency of this parameter in overlapping NAF/SON functions should

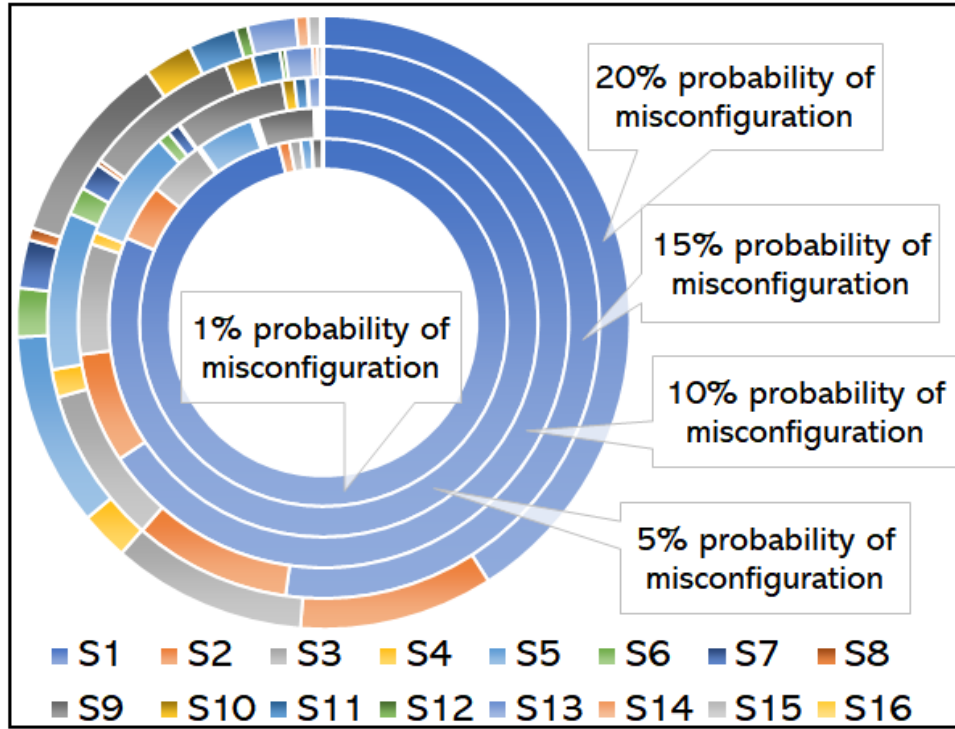


Fig. 4.7: Steady state distributions versus misconfiguration probability.

be less, to minimize the probability of misconfiguration. In reactive network automation functions, the probabilistic distribution of the activation of particular network automation depends upon the mobility and usage behavior of the users. However, in case of proactive SONs, frequency of different network automation functions depends upon the policies defined by the operator. As a result, network automation function scheduling can be viewed as a multi-objective optimization problem to determine the optimal scheduling such that the probability of the system becoming suboptimal is minimized.

Figure 4.6 shows the probability of each state against the number of times a network automation function is activated. With the assumption that the system is in state 1 at time $t = 0$ with the transition probabilities as prescribed in Table 4.3, the results for the state probabilities for 512 transitions of a network automation operation using the Chapman–Kolmogorov equation are shown. It is observed that after approximately 64 transitions, the network achieves the steady state distribution, that is given as:

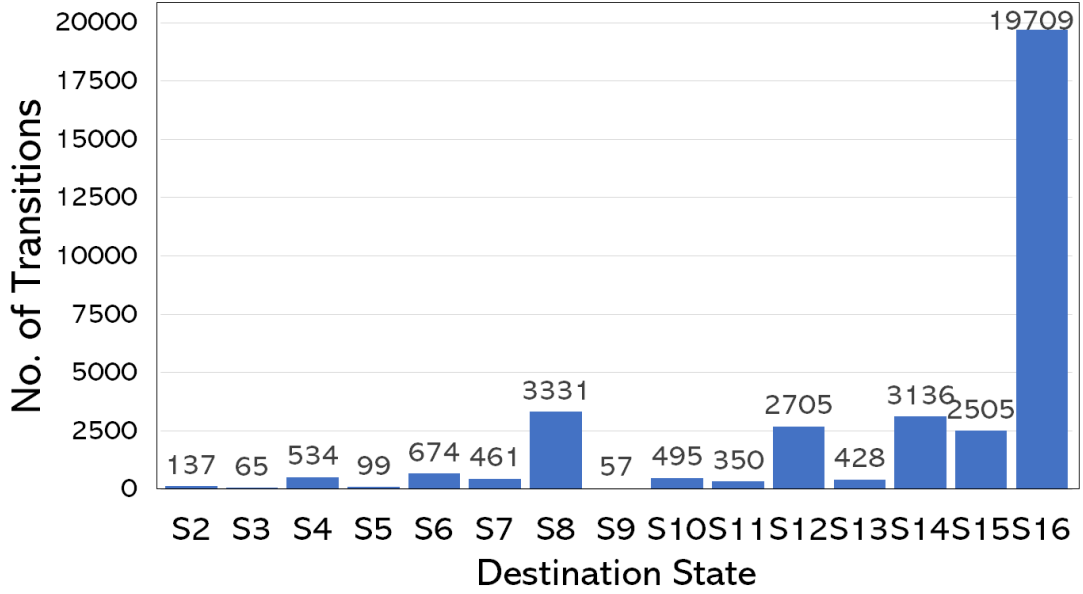


Fig. 4.8: Mean first passage time from state ‘S1’ to other states.

$$[0.88]\pi_{16} = [0.6561, 0.0729, 0.0729, 0.0081, 0.0729, 0.0081, 0.0081, 0.0009, \\ [0.88]0.0729, 0.0081, 0.0081, 0.0009, 0.0081, 0.0009, 0.0009, 0.0001].$$

The fraction of times the network will be in each of the states is not changing after 64 transitions till 512 state transitions. As per the steady state distribution, in the long run, the network remains in the optimal state 65.61% of the time and remains in the suboptimal state for 34.39% of the time. We plot the steady state probability distribution versus the probability of misconfiguration in Figure 4.7. The steady state distributions as defined earlier, is depicting the fraction of time for which the network will be in each of the state in the long term. Figure 8 shows different values of probability of suboptimality. The network will be in a stable state most of the times when there is 1% chance of parameter misconfiguration, i.e., the parameters are configured correctly with higher probability (96.06%) for different activated network automation functions in the network. However, as this misconfiguration probability is increased in steps from 1% to 5%, 10%, 15%, and 20%, i.e., most of the configurations result into transforming the network into suboptimal, the network optimal duration time reduces to 81.45%, 65.61%, 52.20% and 40.96%, respectively. This result demonstrates that higher the parametric misconfiguration probability in a network automation function, higher are the chances

of the network to be in the outage/degraded performance. Additionally, when there is a 15% chance of parameter misconfiguration, approximately half of the time the network remains in the suboptimal state.

Figure 4.8 shows the mean first passage time of DTMC from the optimal state S_1 to all the other suboptimal states $S_2 - S_{16}$, which is calculated using equation (14). The transitions state values vary from 57 (S_9) to 19709 (S_{16}). As the minimum number of transitions from the optimal state S_1 to the suboptimal state S_9 is 57, it can be stated that for the first time after the launch, the network automation functions are optimized by configuring the parameteric values correctly for the initial 56 transitions. However, the network is expected to undergo the suboptimal state immediately after 57 transitions. Thus, using this network performance metric we can predict when the network is going to experience the suboptimal state for the first time.

4.4 Utility of the Proposed Framework

The continued increase from 2G to 5G in the complexity, densification, and the number of the configuration parameters per site lead to a higher risk of the network performance degradation. As to date most of the network parameter optimization is performed manually, and the human error risk increases with the number of configurable parameters and the ultra dense networks. The current design of the network automation functions is prone to a variety of parametric, logical, and other type of conflicts that can cause the suboptimality of the network. Hence, the rate of the under-performance is proportional to the complexity and density of the network. Consequently, to achieve reliabilty, in addition to the physical layer innovations for the increased throughput, emerging network needs resilient, proactive, and predictive fault/outage detection frameworks. The presented framework does not only provides a tool for the reliability evaluation of the given network, but also can act as an enabler for the proactive preventive maintenance of the suboptimal networks. It can help to minimize the detection time of the fault/outage or avoid the performance degradation altogether by preempting them. More specifically, the

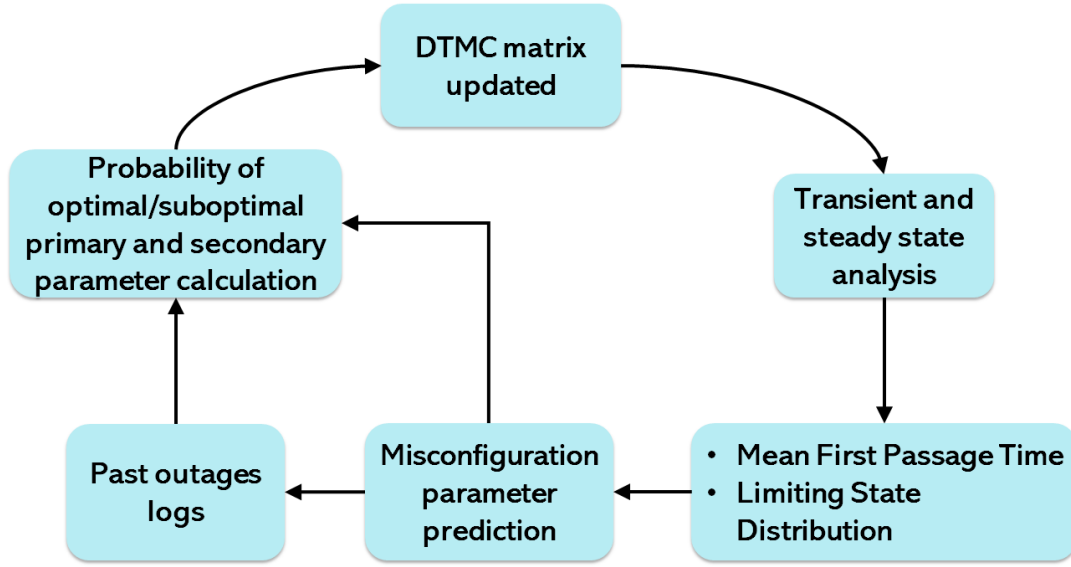


Fig. 4.9: Parameter configuration scheduling prediction framework to avoid manual/network automation conflicts.

proposed framework predicts the expected time of the first occurrence of the fault and long-term reliability behavior of the network, as shown in Figure 4.9. By leveraging the proposed DTMC model and using the standard ML tools for initializing the transition probabilities, operators can prioritize and schedule the proactive network maintenance verification and optimization. The numerical results corroborate the fact that the primary parameters which overlap in the multiple network automation functions or manual optimization routines have the highest probability of misconfiguration. Hence, to enable the predictive and prognostic managements that will reduce the risk of faults, frequent verification of these particular parameters should be scheduled. The results show that with the increasing probability of selection as a primary parameter, the probability of misconfiguration also increases. By leveraging the proposed reliability analysis framework, it is possible to calculate the first occurrence of misconfiguration probabilities for a given parameter to avoid faults/outages. Hence, we can prioritize the verification of each parameter based on its estimated misconfiguration probability.

Thus, the proposed model not only estimates the likelihood of the network transitioning into a suboptimal state, but also enables operators to evaluate the expected time for entering and persisting in such states. This temporal insight supports the development

of a proactive fault detection framework that complements existing outage compensation mechanisms, facilitating end-to-end proactive self-healing in 5G and beyond (5GB) networks [209, 212]. By triggering preemptive recovery actions as soon as a degradation is predicted, the model allows early intervention before service-level objectives are violated. Consequently, this approach offers significantly enhanced network reliability compared to conventional reactive fault management, paving the way for predictive, zero-touch automation in future ultra-reliable network infrastructures.

4.5 Conclusion

In this chapter, we proposed a proactive fault prediction framework based on Discrete Time Markov Chain (DTMC) to model the probabilistic reliability behavior of Network Automation Function (NAF)-enabled cellular networks. By assuming geometric distributions of failures, the framework effectively captures the impact of NAF/SON parameter misconfigurations and conflicts—one of the critical yet often overlooked causes of network degradation in ultra-dense, automated infrastructures. Unlike prior reliability studies that focus primarily on static cell configurations or structural factors, our model incorporates dynamic parametric interactions and evaluates their influence on both transient and steady-state reliability behavior. The proposed approach not only predicts the probability of the network entering a suboptimal state but also estimates the expected time to first degradation and steady-state performance levels, enabling preemptive intervention. This allows operators to proactively activate self-healing or compensation strategies before an outage actually occurs, thereby minimizing downtime and supporting the stringent reliability demands of emerging 5G and beyond (5GB) applications. Numerical results further validate the effectiveness of the model in identifying sensitive parameters, quantifying fault impact, and optimizing multi-parameter configurations under realistic scenarios. By enabling conflict-aware orchestration of NAFs, this framework provides a foundational step toward zero-touch reliability assurance and proactive fault management in next-generation networks. Future extensions of this work will focus on

estimating DTMC transition parameters using real-world MDT logs and failure data. Additionally, integrating learning-based techniques such as Reinforcement Learning or Bayesian Networks will further refine the model's predictive agility, enabling closed-loop, adaptive fault management under evolving network contexts.

CHAPTER 5

An Intelligent Anomaly Detection Solution to Secure Multi-purpose Future Networks

5.1 Introduction

5.1.1 Motivation

The Internet of Things (IoT) has transformed the landscape of communication, enabling seamless connectivity across domains such as healthcare, smart cities, and industrial automation. Cellular IoT networks like LTE-M and NB-IoT serve as the backbone of this connectivity, offering extended coverage, privacy, and real-time analytics capabilities. With the proliferation of these technologies, the number of IoT devices is projected to reach 75.44 billion by 2025 [213], expanding the reach of digital intelligence to “anything, anywhere.” To meet these demands, cellular networks must support ultra-reliable, low-latency communication—especially for mission-critical applications such as outbreak monitoring, autonomous healthcare diagnostics, and public safety systems. A key component in enabling automation within these networks is the Minimization of Drive Tests (MDT), standardized in 3GPP Release 10 [214]. MDT enables user equipment (UE) to report critical Key Performance Indicators (KPIs) such as GPS coordinates, RSRP, RSRQ, SINR, CQI, and other environmental measurements [215, 216]. These reports are leveraged not only for Zero-Touch SON functions such as base station optimization and power allocation but also for high-stakes AI applications in epidemic forecasting and mobility-aware healthcare analytics. However, while MDT data enables real-time visibility into network performance, its reliance also introduces significant security vulnerabilities. These reports, which carry sensitive user and network information, are susceptible to adversarial manipulation. Malicious actors can inject false data, either by spoofing devices or forging CQI reports to trigger incorrect SON actions like unwarranted power

scaling or compensation strategies, thereby degrading performance and potentially triggering outages. Recent attack scenarios show that even minimal spoofed MDT data can falsely suggest coverage blackspots or overload conditions, thereby impacting the network’s optimization loop. This becomes especially dangerous when real-time MDT data feeds into mission-critical AI systems, for example, outbreak monitoring applications that predict disease spread based on crowd-sourced mobility traces. In such contexts, compromised data not only degrades network performance but misleads public health interventions. A single injection of falsified location data can trigger false hotspot alerts, redirecting medical resources or causing unjustified lockdowns. Worse still, once these predictions are undermined, the trust in AI-driven healthcare solutions is jeopardized, leading to long-term systemic consequences.

Although existing frameworks attempt to detect performance anomalies using reactive and real-time monitoring, such approaches fall short when facing stealthy data tampering attacks. They are unable to respond swiftly, leaving the system in a degraded state for hours unacceptable in systems demanding five-nines (99.999%) reliability. This calls for a paradigm shift from reactive detection to proactive, AI-powered security mechanisms that can safeguard the reliability of MDT-driven intelligence at both network and application layers. Moreover, as highlighted in prior works [217–220], IoT systems are uniquely vulnerable to security threats such as eavesdropping, identity spoofing, data forgery, and denial-of-service (DoS) attacks. These challenges remain insufficiently explored when considering their cascading impact on downstream decisions enabled by AI/ML. Unfortunately, most studies treat IoT security as a standalone issue, ignoring its ripple effects on autonomous decision-making processes and public trust in such systems.

To bridge this gap, this chapter introduces a dual-pronged AI security framework that not only defends against data integrity breaches in MDT streams but also ensures that AI models relying on such data remain trustworthy and robust. By integrating AI-driven detection with real-time verification mechanisms, the proposed framework serves both network operators and healthcare agencies, enabling resilient anomaly detection, secure data fusion, and proactive epidemic response. This ensures that future 6G-enabled net-

works can serve as the foundation for secure, scalable, and autonomous systems, capable of managing both network and societal-level threats in a unified, intelligent fashion.

5.1.2 Relevant Work

The security and privacy preservation of IoT enabled networks is a formidable challenge due to the diversity, complexity, and enormous interconnected resources. There are numerous sophisticated attacks that an adversary can perform on IoT network such as eavesdropping, modification, tampering, and repudiation, due to the lack of efficient tracking mechanisms, this false information may lead to users' inconvenience and serious consequences [218–220]. Based on the vulnerabilities in the sensing, network, and application layer in IoT networks, the attack can be classified into four categories [220]:

- **Physical attacks:**

In physical attacks, the adversary targets the hardware of the devices. The adversary to identify new vulnerabilities, access the device by buying a copy of the IoT device and performs the reverse engineering and by creating a false attack to observe/analyze the estimated outcomes [221].

- **Network attacks:**

In network attacks, the malicious node tries to compromise the connectivity between the device or the connection between device and cellular network.

- **Software attacks:**

In software attacks the adversary focuses on the denial of services and data stealing by employing the spyware, viruses or worm.

- **Encryption attacks:**

In encryption attacks, the focus is to destroy the encryption technique, to obtain the private key of the network.

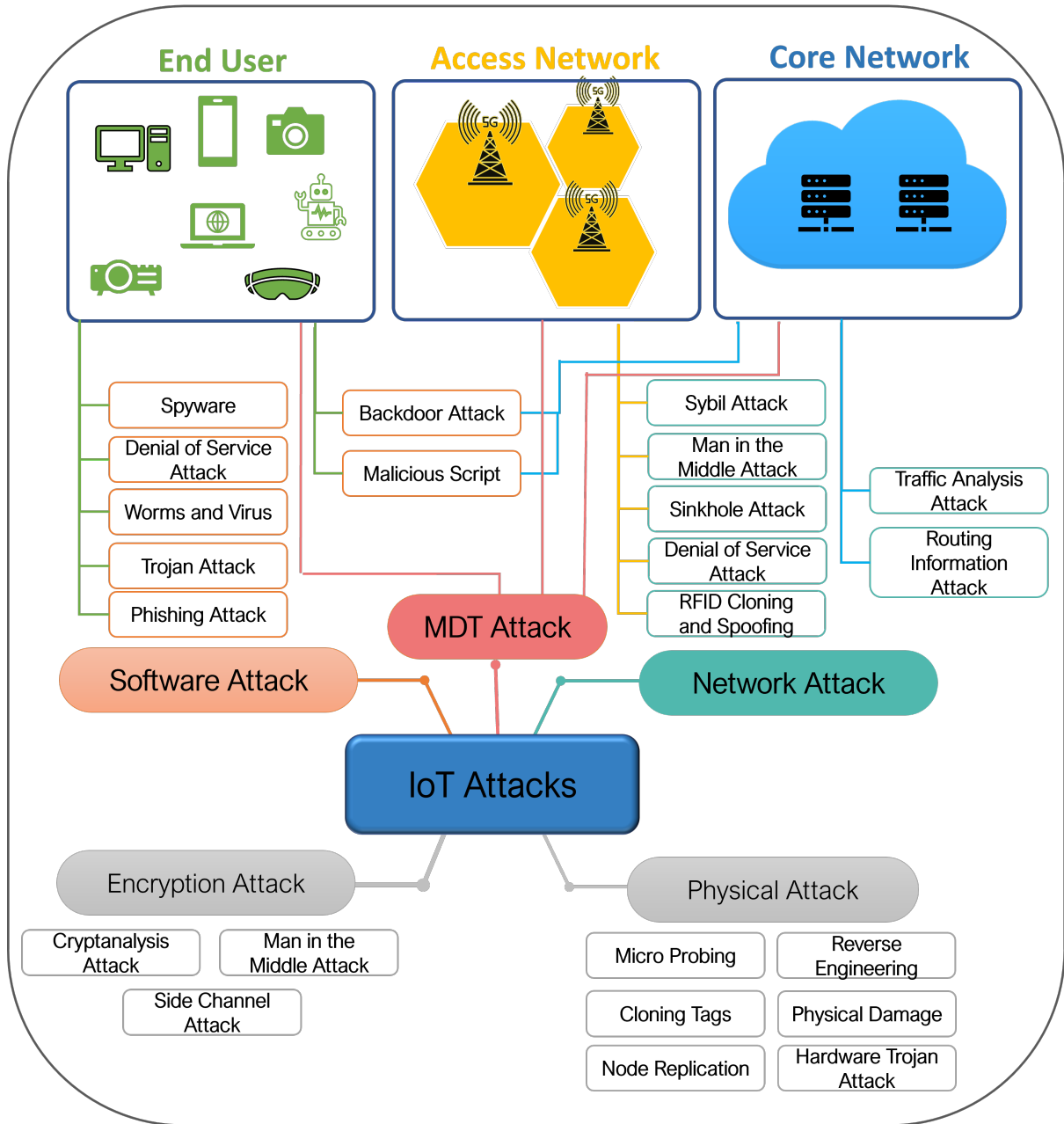


Fig. 5.1: Classification of security threats in smart systems.

Based on the vulnerability, occurrence, damage level, and detection chances in various network segments (UE, access network and core network), we have summarized various types of network and software attacks encountered by the IoT devices, as shown in Figure 5.1.

Network Attacks

In such attacks, a malicious entity attempts to disrupt the communication or connectivity between devices and the network infrastructure. Further types of network attack includes: *Sybil attacks*: The malicious node illegitimately claims to have multiple identities and manipulates the sensors and ad-hoc networks by impersonating other nodes. *Traffic analysis attack*: The adversary gets the network information by intercepting and examining the node messages. *Man in the middle attack*: The attacker obtains the information being exchanged between the two IoT devices by eavesdropping. *Routing information attack*: The malicious node induces complexity in the network by tampering, spoofing, or modifying routing information, consequently the information is forwarded to the wrong target network or the packets are dropped. *Denial of service attack*: The attacker can mislead the smart devices connected to an authenticated wireless network to disconnect the supply of mobile broadband connections and link to a false network, which leads to the discontinuation of services. *Sinkhole attack*: Also known as blackhole attack, the node falsifies the routing information to the neighboring nodes so that all the data is routed towards the compromised node. The aim is not to forward any data, but only creating a sink. *Radio Frequency Identification (RFID) cloning and spoofing*: The attacker copies the data from the pre-existing RFID and injects the wrong data by the cloned nodes. In spoofing on the other hand, the adversary spoofs RFID signals then sends false information which is accepted by system as correct.

Software Attacks

Software attack have several types such as: *Spyware*: The adversarial node uses malicious code that has the ability to replicate and spread in the system through email attachments, and downloading harmful files from the Internet. *Worms and viruses*: Without the need of user interface, worms can transport their copies to other device using a particular transport mechanism and infecting the system. While in case of viruses, replications of files are made by consuming the system capacity, it leads to unknown source of installation

without user permission and infect the user files. *Trojan attack*: Trojan can sneak into the system without informing the administrations. The attacker disguises malware as a legitimate piece of software (i.e., program or app) or even harmless documents to trick users into willingly installing to their computer.

Various IoT attacks mentioned above can target different network segments, i.e., the target of some network and software attacks are the end user devices, whereas other network and software attacks are specifically designed for the access and core networks as mentioned in Figure 5.1. Moreover, there are some attacks that can be launched simultaneously in more than one network segment, making them difficult to track, compared to the other attacks. For instance, backdoor attacks, DoS attacks, and malicious script attacks can co-exist in multiple network segments, making it harder to identify the attack and launch the defense mechanism, accordingly. Numerous countermeasures are proposed in the literature for the network and software attacks [220, 222–225]. In the context of cellular IoT networks, one form of attack we introduce in this work is “MDT attack”. In contrast to other types of IoT attacks having an impact on either one or two network segments, this newly introduced MDT attack has its deleterious impacts on several network segments. For instance, when the adversary takes charge on the registered IoT device and start sending malicious MDT/CSI reports to send false data to the base station (BS), hence attacking the access network. When the SON engine performs the network optimization based on the misleading MDT/CSI reports, the impact of this attack may propagate to wider radio access network (RAN) as well as the core network. Hence, the disseminating nature of this attack in all the three network segments, makes it distinctive and more virulent compared to other types of attacks.

In the context of crowdsourced epidemic monitoring, MDT attacks pose an equally significant threat. Adversaries can falsify MDT data, such as user locations or mobility traces, leading to inaccurate epidemic spread predictions. This misinformation can trigger unnecessary public health measures, such as lockdowns or resource reallocations, which not only disrupt societal functions but also undermine the trust and credibility of epidemic monitoring systems. The dual impact of MDT attacks on both cellular network opera-

tions and digital healthcare applications emphasizes the critical need for robust security mechanisms to ensure the integrity of MDT data in these contexts.

5.1.3 Contributions

Specifically, the contributions of this chapter are summarized as follows.

- We introduce an impactful adversarial MDT attack and analyze its cascading detrimental effects on both cellular network performance and crowdsourced epidemic monitoring.
- We address the repercussions of adversarial attacks on Zero-Touch automation functions, demonstrating how fake MDT/CQI reports create conflicts among self-optimizing network (SON) functions, such as cell outage compensation (COC), undermining the system’s optimization capabilities. Additionally, we analyze how falsified MDT data can lead to inaccurate epidemic spread predictions, triggering unnecessary public health responses, such as lockdowns.
- To detect and mitigate adversarial MDT attacks, we propose a novel multi-module Malicious MDT Reports Identification Framework (MRIF), which utilizes machine learning techniques to distinguish false MDT reports from authentic ones, safeguarding both cellular network operations and healthcare applications.
- We demonstrate the effectiveness and robustness of the proposed defense mechanism through a detailed use case involving the cell outage compensation (COC) SON function. Experimental results validate the framework’s capability to protect network performance and ensure reliable data for epidemic monitoring.

5.1.4 Chapter Organization

The remainder of the chapter is organized as follows. We introduce our devised adversarial MDT-based attack and its detrimental affect in Section 5.2. Section 5.3 presents

the MRIF framework as a countermeasure of adversarial MDT attacks. Meanwhile, Section 5.4 describes the system model and the case study as the analysis of the effectiveness of the proposed solution. The open research challenges are discussed in Section 5.7. Finally, the chapter is concluded in Section 5.8.

5.2 Adversarial MDT Attacks

In this section, we describe about Minimization of Drive Test (MDT) reports, the information that is provided by MDT and how an adversary can be responsible for sending compromised MDT reports to the network creating adversarial MDT attacks to have a cascade of detrimental impact and holistically jeopardize the network performance.

Minimization of Drive Test

MDT reports are vital for network monitoring and optimization, as the IoT and other digital connected user devices periodically update the network about the user mobility, signal condition, providing information of spatial and temporal data related to the quality of experience (QoE) in a specific area. Through MDT measurement configurations, network can instruct the UEs to log network measurements, such as location coordinates, RSRP, RSRQ, and SINR of the serving and neighboring cells, and send them to the serving BS through Radio Resource Control (RRC) signaling messages. Thus, MDT eliminates the need of physical drive test for the system quality assessment. Furthermore, compared to drive test based performance assessment that can gauge performance only on paved roads, MDT reports can not only provide more spatially holistic picture of the network performance but also they offer opportunity for more timely coverage analysis, user experience, and anomaly detection in the network performance. The MDT scheme supports two reporting configurations:

1. **Immediate Mode:** It is reporting wherein the user device instantly reports the measured radio conditions when the preconfigured triggers are met, and

2. **Logged Mode:** It is the mode wherein the user device stores measurements and reports them upon the expiration of a periodic timer.

MDT reports are tagged with location information of the reporting UEs. These attributes play a significant role for facilitating the network to take necessary actions for the optimal network performance [226].

5.2.1 Vulnerability of Network Performance to MDT-based Attacks

The true potential of SON is contingent to the timely and accurate availability of network MDT measurements. Based on the intelligence extracted from the reports, SON engine initiates appropriate SON-functions to achieve optimum network performance that aligns with the business tailored objectives prescribed by the operator. Effectively, the MDT feature is a pre-requisite for enabling a range of diverse 3GPP standardized SON functions such as mobility load balancing (MLB), cell outage compensation (COC), coverage and capacity optimization (CCO), enhanced inter-cell interference coordination (eICIC) and Energy Efficiency (EE) to name a few [227]. However, there lies one caveat here - if we look from the security perspective, SON engine optimizes the network by relying on the information sent by the UE which is in direct reach to the public. This inherently can serve as a backdoor for the hackers to influence the SON mechanism by sending modified/false MDT measurements within UEInformationResponse RRC messages back to the network through the compromised/hijacked UEs. Even the sophisticated UE devices such as smartphones, which have additional security softwares, have vulnerability due to the availability of open source baseband stack like OsmoComBB [228], that can be run on compatible phones. OsmoComBB allows obtaining of general understanding of the structure of cellular stack, which can be very helpful for binary analysis and reverse engineering of baseband stacks, commonly available in the market. Compared to smartphones based UEs, however, it is the inexpensive IoT devices having limited intelligence and no additional security layer, which makes them more vulnerable to the MDT-based attacks, thereby providing core motivation for this study. For conciseness, here forth, we will use

the term UE to represent all type of cellular and other devices including IoT. Hackers can exploit these vulnerabilities of UE to improvise cellular network signaling attack. More specifically, in the context of MDT adversarial attack, hackers can modify the baseband stack to embed deceiving measurements in the UEInformationResponse RRC messages and send them back to the SON engine of the network. If the malicious UE succeeds in modifying protocol stack of the baseband processor, then a successful attack can be executed, as state-of-the-art SON engines are designed to trust whatever data it receives from that UE in the network.

5.2.2 Repercussions of MDT/CSI-based Attacks

If a network attacker succeeds in modifying the network stack in a smart digital device then they will be capable of sending customized false MDT reports back to the SON engine. These deceiving reports can force SON engine to erroneously trigger several SON functions that can have devastating impact on the entire network performance. For instance, hacked UEs can falsely report MDT reports with very low RSRP values from areas which actually have good coverage, thus can trigger COC SON function in the network when and where it is not needed. This may lead to an unstable network condition forcing the network to drift into a sub-optimal configuration state, as shown in Figure 5.2 and network parameter are mentions in Table 5.1. The figure shows that in case of actual outage detection Figure 5.2 a), the COC SON function is triggered and increases the transmit power of neighboring cells to compensate the UE previously connected to the cell in outage as shown in Figure 5.2 b), hence the system SINR is restored to quite an extent, as demonstrated in Figure 5.2 c). However, if outage compensation mechanism is activated based on malicious MDT reports, (Figure 5.2 d and e). The SON function is triggered erroneously, causing a decrease in the network SINR because of the increase in the interference caused by the unnecessary increase of transmit power of the compensating BS. This negative affect of MDT-based attack that leads to erroneous triggering of COC by SON engine can be seen in Figure 5.2 f). This use-case exhibits how an attacker by sending the false MDT report to a SON engine, can mislead SON engine to erroneously

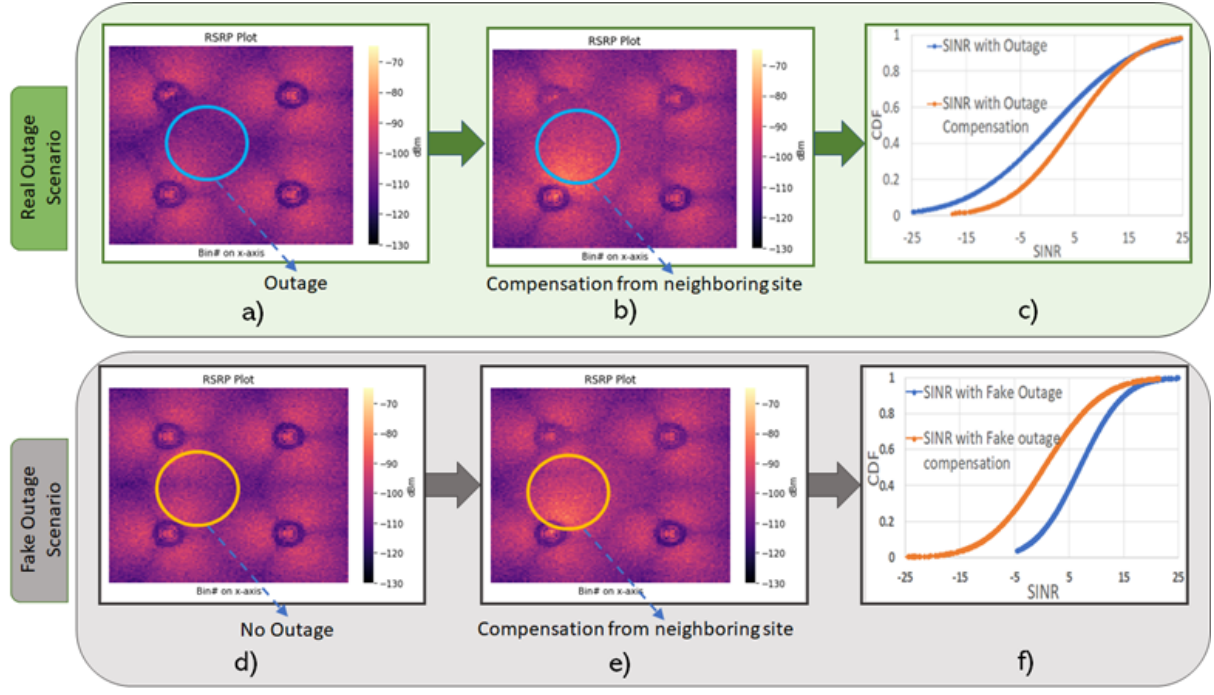


Fig. 5.2: Adversarial MDT attack impact on the system's SINR and network performance.

TABLE 5.1. Description of simulation parameters.

Parameter	Description	Value
Simulation area		1km ²
Operating frequency		2100MHz
Macro cell transmit power		35dBm
Macro cell height		30m
Number of sites in outage		3
Pathloss exponent		4
Shadowing standard deviation		6.9
User density		1000

trigger/halt a SON use-case and thus jeopardize the network performance.

An adversarial attack of this kind can target any SON function as shown in Table. For instance, a compromised UE can send malicious reports related to handover parameters that can erroneously commence Mobility Robustness Optimization (MRO) SON function. Same is true for other SON functions such as MLB and EE. In fact, a specially crafted coordinated attack by the compromised UEs can initiate any desired SON function in the network. Table 5.2 shows the potential impact of the adversarial attack on the 3GPP defined SON functions and identifies the associated negative impact on the network KPIs. We categorize the SON functions which are most likely to be negatively affected from the

TABLE 5.2. Implication of MDT attacks on SON Functions.

SON Function	Potential Attack Description	Affected KPIs	Severity of Effect when Attacked
MLB	Adversarial UEs can report poor SINR and request for higher throughput. To accommodate the request, BS will allocate more Physical Resource Blocks (PRBs) which might trigger false MLB	• PRB Utilization • Traffic Volume • MIMO Rate • Carrier Aggregation Activation • Accessibility • Handover Success Rate • Downlink Throughput	High
MRO	Handovers mainly depend on the RSRP measurement of the source and target cell. An attacker can trigger too early or late handovers, leading to increased handover failures. As a compensation, MRO might add unreasonable CIOs	• Handover Success Rate • Downlink Throughput • Retainability • Integrity	High
COC	Fake outage can cause serious issues in the network if undetected. The BS might activate COC to increase the power or change the tilt of the BSs to cope with outage, which affect the SINR of the BS under attack	• Accessibility • Retainability • Handover Success Rate	Medium
CCO	The adversarial UE can report incorrect measurement, e.g. very poor RSRP or RSRQ. In return, BS increase its coverage which might increase interference and overshooting with other neighboring cells	• Coverage • Capacity • Handover Success Rate	Medium
eICIC	The malicious reported RSRP or RSRQ can falsely trigger the eICIC function. Based on that the BS can unnecessarily change the coordination of physical resources between neighboring cells to reduce interference	• Capacity • Integrity • PRB Utilization	Medium
EE	Some BS are configured to turn off during certain period (non busy hours) to conserve energy. An attacker might fool these BSs and not letting it sleep, thinking some UEs need connection by sending false request or measurements	• Energy Efficiency • Availability	Low

MDT attacks. In case of centralized SON, the local impact of MDT attacks can cause interactions at central SON server that can have an influence on the global behavior of the network. In case of distributed SON, in addition to local performance degradation, the unnecessary call for SON functions caused by MDT attacks can lead to the wastage of network resources like computational power, signaling bandwidth, etc. Moreover, it has been highlighted in the literature that multiple uncoordinated SON functions are subject to a large number of potential logical, parametric, or objective conflicts [229,230]. Therefore, even with distributed SON, local MDT attack can lead to the sub-optimal performance at a larger scale. Furthermore, coordinated attacks by compromised UEs can be configured such that the network never reaches to its optimum stable point, rather it keeps on oscillating between different states having negative impact on the operational expenses (OPEX) and QoS. Therefore, it is crucial to devise a robust defense mechanism against these MDT-based adversarial attacks.

The motivation behind selection of Cell Outage Compensation (COC) as a use case for the detection of fake outages and MDT attacks. MDT-based coverage and performance estimation is a fundamental step to enable the AI-based zero automation i.e., the advanced

self-configuration, self-optimization, or self-healing routines. Hence, it is a crucial aspect for several SON functions as listed in Table 5.2. In this table, the SON functions are ordered in terms of the severity of the impact of MDT attacks on these functions. While Mobility Robustness Optimization (MRO) and Mobility Load Balancing (MLB), when compromised by MDT attacks, have more severe impact on the network and users due to the multiple Key performance Indicators (KPIs) that might get affected, attacks on the COC function is very crucial as well. Additionally, since the fake outages involve physical parameters (i.e., tilt, azimuth, transmit power) rather than soft parameters used in MLB or MRO (i.e., CIO) it can easily create a visualization, through coverage maps of how the attacks are being done, how the undesired compensation takes place and what will be the impact of after the compensation. This visualization will not be possible in the case of MLB or MRO, since their impact cannot be observed using coverage maps. Although, the use case selected is on fake outages, the framework is designed to work with any type of potential attacks on any of the SON functions listed in Table 5.2, including MLB and MRO functions as discussed. Only the anomaly detection module of the framework will require retraining specific to use cases to detect anomaly in KPIs associated with use cases. In case of MLB and MRO, these KPIs include congestion indicators, QoS indicators, handover success ratio, radio link failure and ping pong to name a few.

5.3 MRIF Framework to Counter MDT Attacks

This chapter proposes a defense mechanism called Malicious MDT Reports Identification Framework (MRIF) to detect the deceitful intruders and block the adversarial MDT attacks. We will discuss the two modules of MRIF; where the first module Anomaly Detection Module (ADM) detects any anomaly from the normal MDT reports based on machine learning algorithms (XGBoost and Autoencoder). And the second module that we categorized as Malicious Reports Filtering Module (MRFM) which further segregates the genuine outage from the fake outages reported by the malicious intended MDT reports.

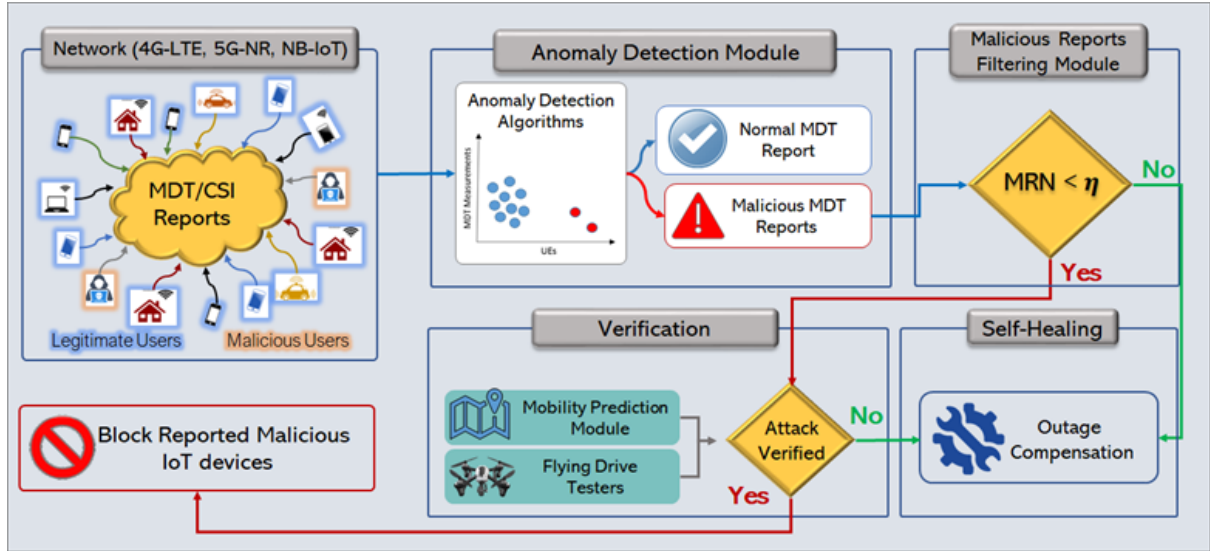


Fig. 5.3: Malicious MDT Reports Identification Framework

Adversarial attacks have the tendency to take control over the IoT devices and propagate the impact of the attack into different networks segments by sending false MDT reports, leading to sub-optimal performance. Fake MDT reports can be logged either from the registered and unregistered end-user devices which makes the detection challenging. This inherent vulnerability of the network demands the need of devising countermeasures against such pernicious attacks. In this section. The breadth and depth of MRIF anchor on its capability to detect outages and distinguish fake MDT/CSI reports from the real outage reports. The framework consists of the following modules as shown in Figure 5.3

5.3.1 Anomaly Detection Module (ADM)

This module receives the MDT reports from various users and detects any anomalies in the reported MDT/CSI data. One example of such anomaly which attackers can exploit is the cell outage. ADM leverages ML-based anomaly detection algorithms such as Cell Outage/Degradation Detection SON algorithms [231]. Anomaly detection algorithms are commonly based on classification, nearest neighbor, clustering, and statistical measurements, to detect and identify the anomalous MDT reports that imply a degraded performance due to some possible fault in a cell. Some of the techniques traditionally used for anomaly detection are One-Class SVM and LOF [232–234]. Recent studies [234, 235]

also demonstrate that leveraging XGBoost [236, 237], Autoencoder [238–240] gives better outage detection and more resilience to various scenarios. For our proposed case study, we have considered the XGBoost, and AE. Leveraging these models enable the separation of normal MDT reports from the outage data, with satisfactory accuracy.

5.3.2 *Malicious Reports Filtering Module (MRFM)*

To further filter out the malicious MDT reports indicating the false outage from the genuine MDT/CSI reports indicating true outages, the outage data is fed into the MRFM. The working of this module is premised on the notion that the malicious reports from a compromised UE will differ from the genuine anomalous MDT reports which are in the same vicinity. This observation stems from the fact that the shadowing phenomenon has a significant impact on the reported MDT results, particularly in an environment where it remains correlated over small distances. Therefore, if the number of malicious reports in the neighboring (MRN) region is greater than a threshold it indicates actual outage which might be due to technical or operational discrepancy in the cell or network. A real outage cannot be an isolated event for single or few users in a particular region, rather a significant number of users from the same region should report it as the outage event. In the LOF scheme [234, 241, 242], one hyperparameter, i.e., $n_{neighbor}$ η (threshold) is essentially capturing the essence of this observation. Hence, after performing the hyperparameter tuning, we are able to get the good performance for $\eta = 10$, i.e., if less than 10 neighbor UEs report poor MDT in a particular region then it manifests an adversarial attack, which depending on the severity will either require an additional verification process or immediate blocking of the compromised adversary sending false MDT/CSI reports.

MRIF-Enabled Secure Mobility Monitoring Verification of adversarial attacks using Flying Drive Testers

In addition to the multi-module MRIF defense mechanism, an additional layer of protection can be introduced by leveraging Flying Drive Testers (FDTs) to enhance the

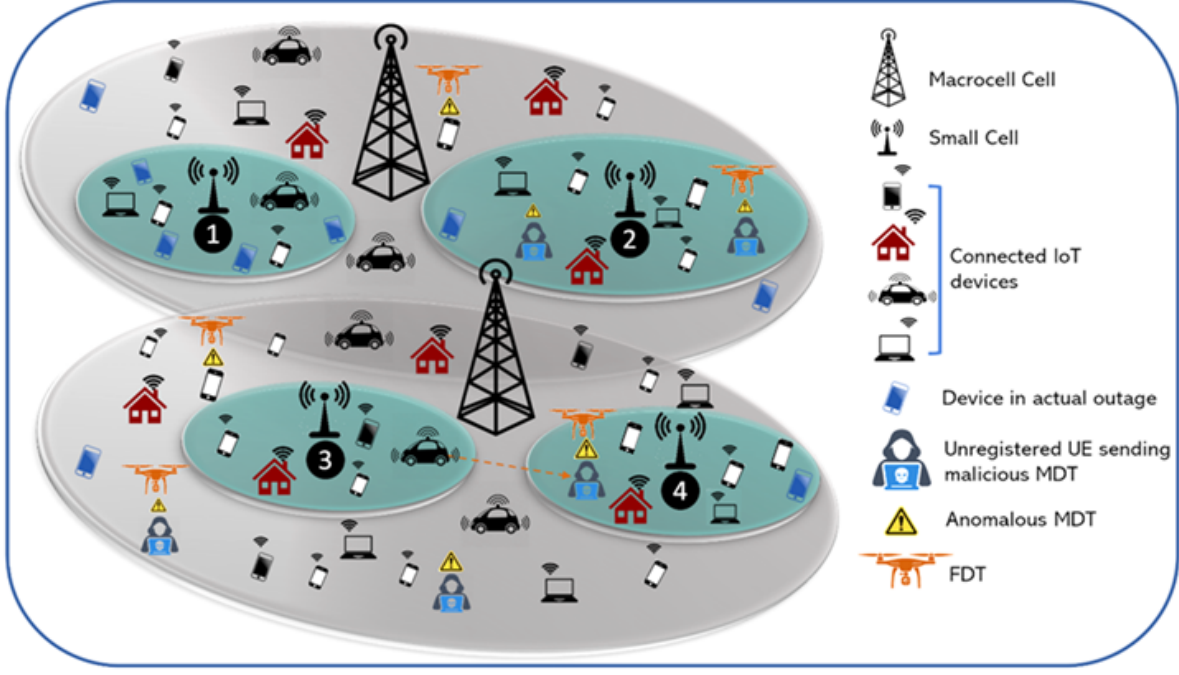


Fig. 5.4: Illustration of false MDT injection attacks and MRIF-based mitigation through cross-validation using FDT feedback.

robustness of the framework against MDT-based adversarial attacks as shown in Figure 5.4. This figure demonstrates how MRIF operates to counter MDT-based attacks in mobile networks. In Cells 2 and 4, malicious users inject fake MDT reports, falsely indicating network outages. While traditional systems may react to such inputs, MRIF applies multi-stage filtering to verify anomalies. In Cell 4, the framework cross-validates reports with feedback from trusted IoT devices in Cell 3 and Field Drive Test (FDT) data. This layered analysis helps detect and discard fabricated MDTs in real time, preserving network stability and maintaining trust in automated decisions—especially critical for latency-sensitive and safety-critical services.

5.4 Results and Discussion

In this chapter, we present the performance evaluation of Malicious MDT Reports Identification (MRIF) framework to detect the malicious MDT reports. Specifically, we leverage three machine learning algorithms including Autoencoder, XGBoost, Local outlier factor.

We demonstrate the working of the proposed multi-module defense mechanism with a

TABLE 5.3. Description of simulation parameters for outage scenario.

Parameter Description	Value
Simulation area	1km ²
Operating frequency	2100MHz
Macro cell transmit power	35dBm
Macro cell height	30m
Number of macrocells	9
Number of sites in outage	1,2,3
Pathloss exponent	3
Shadowing standard deviation	4
User density	2500,5000,7500

use-case of identifying the adversarial MDT attack in the presence of real outage MDT reports. We aim to demonstrate that the ML-enabled framework can identify fake MDT data sent by the adversary IoT devices from the genuine outage reports. The subsequent discussion and results show the efficacy of the proposed MRIF in detecting the malicious MDT-based attack:

5.4.1 System model and analysis

We utilize a 3GPP compliant system-level simulator SyntheticNet [243] to model the network and generate the MDT data for the outage detection. The relevant parameters are also set according to the standards, given in Table 4.1.

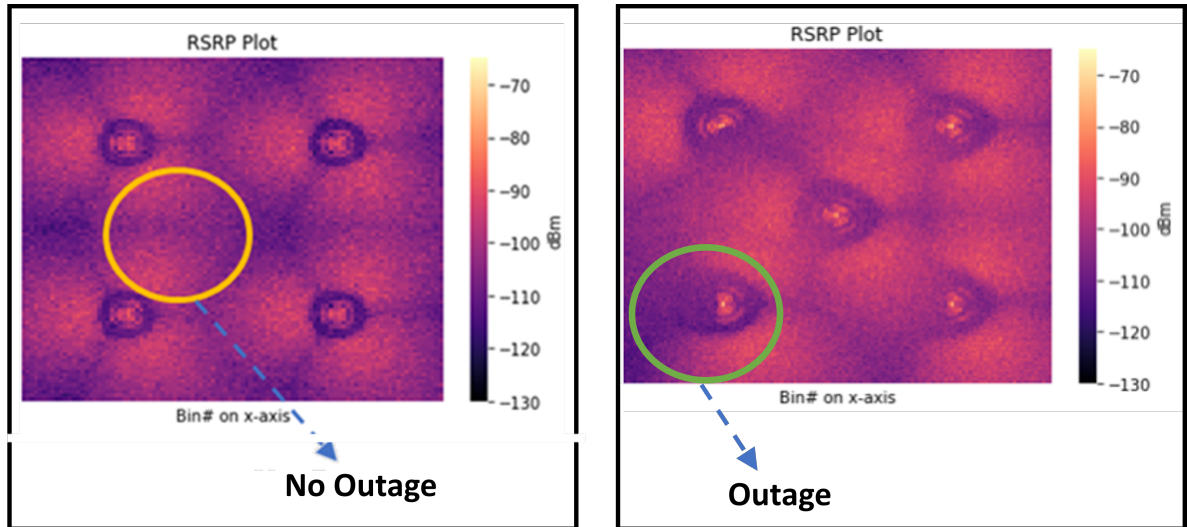


Fig. 5.5: System model with multi-cite cells where some sites are in outage.

The outage scenario is simulated by setting the transmit power of three BSs to zero. The resulting network coverage map with 3 BSs in outage as shown in Figure 5.5. In the area of $1km^2$, three different user density is deployed (2500, 5000, 7500) where the UEs associated to these 3 BSs report real outage by sending low RSRP values in their MDT reports. Whereas, to simulate the adversarial attack we reside to the idea that any user can act as a potential adversary and falsely reports poor signal condition, despite being connected to the normal BS. Thus, the MDT data is comprised of two main categories, i.e., normal MDT reports and outage MDT report. The category of outage reports further constitutes of **i) real outage**, and **ii) malicious outage MDT reports**. The MDT measurement report constitute of the features, i.e., longitude and latitude of the users, RSRP, and RSRQ of the serving base station. To incorporate the correlation information from the neighboring cells we have also included the RSRP and RSRQ from the six strongest neighboring BSs as features.

5.5 Analysis and results for Anomaly Detection Module

5.5.1 Anomaly Detection Module with Autoencoder

For autoencoder training and testing, we generate labeled synthetic MDT data using SyntheticNet [243]. For real implementation of the proposed solution either a simulator or a digit twin of the network combined with the real MDT reports can be used. The data having features (RSRP and RSRQ from the six strongest neighboring BSs) is fed into the ADM with the goal of developing an optimal model that can distinguish the normal reports from the malicious MDT reports. Among the aforementioned ML-based algorithms, we implement the AE and XGBoost anomaly detection due to the robustness, and sensitivity to the higher shadowing data. Following are the details of autoencoder implementation:

- The autoencoder model consists of 3 layers, i.e., 6 by 3 by 6 neurons in each respective layer. It is trained on the normal MDT data and tested using the composite

data containing normal and outage MDT reports. For the generation of accurate results and robustness of the model, we have utilized the 5-fold cross validation to and L1 regularization to further ensure that model is not biased towards any class. It is worth mentioning that while the class labels are not employed in the training of AE, during the testing process, the labeled data is required to evaluate the prediction accuracy.

- When AE is used for anomaly detection, one of the key parameters affecting its output is the choice of the reconstruction threshold. The reconstruction error is used to determine the class of the output in the testing data. In this study, the testing data instances where reconstruction error values are greater than the threshold are considered outages. Hence, selecting the correct representing value of the reconstruction threshold plays a significant role in ensuring accurate detection.

Several methods of various complexity exist for determining the reconstruction error. Manual selection by observing the distribution of reconstruction error is the easiest method, provided it clearly separates the classes [244]. On the other hand, machine learning techniques can also be used to estimate the threshold in complex settings, depending on whether class labels are available or not. Some algorithms determine this value using the mean and standard deviation of the reconstruction error [245].

However, in this study, we resort to a statistical measure, the q th quantile of the distribution of reconstruction error, where parameter q is determined by the ratio, i.e., the number of outage instances to the total instances of training or test datasets [246, 247]. It should be noted that the actual value of the threshold is not insightful, without the knowledge of the range and distribution of reconstruction error values. Hence, this approach of pegging the threshold to the q th-quantile of reconstruction error and making q adaptive to the outage instances count enables the framework to work effectively for varying outage severity scenarios.

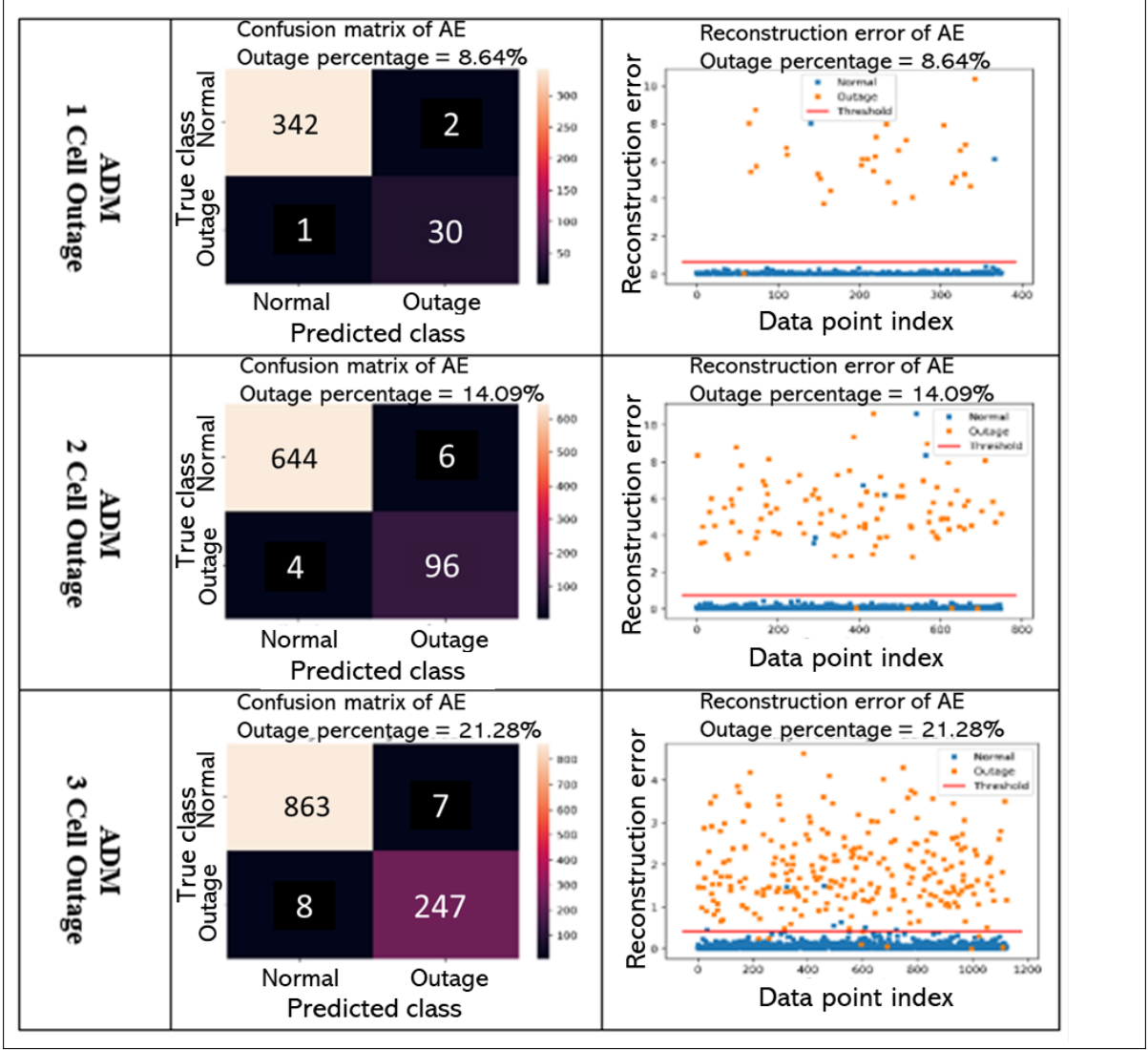


Fig. 5.6: Confusion matrices and reconstruction error plot of Autoencoder for the selected combination of MDT data density, and outage severity.

- For AE training and testing, lowering this threshold may result in the detection of more anomalies, but it may also cause normal instances to be misinterpreted for malicious ones i.e., higher false positive rate. On the other hand, raising this threshold prevents false positive detection at the expense of detecting lesser anomalies. As mentioned earlier, for our anomaly detection framework, we employ the q^{th} quantile of the distribution of reconstruction error. This method of tying the threshold to the q^{th} quantile of reconstruction error and making q adaptable to the number of outage occurrences helps the detection of outages with varying number of MDT reports as shown in Figure 5.6.

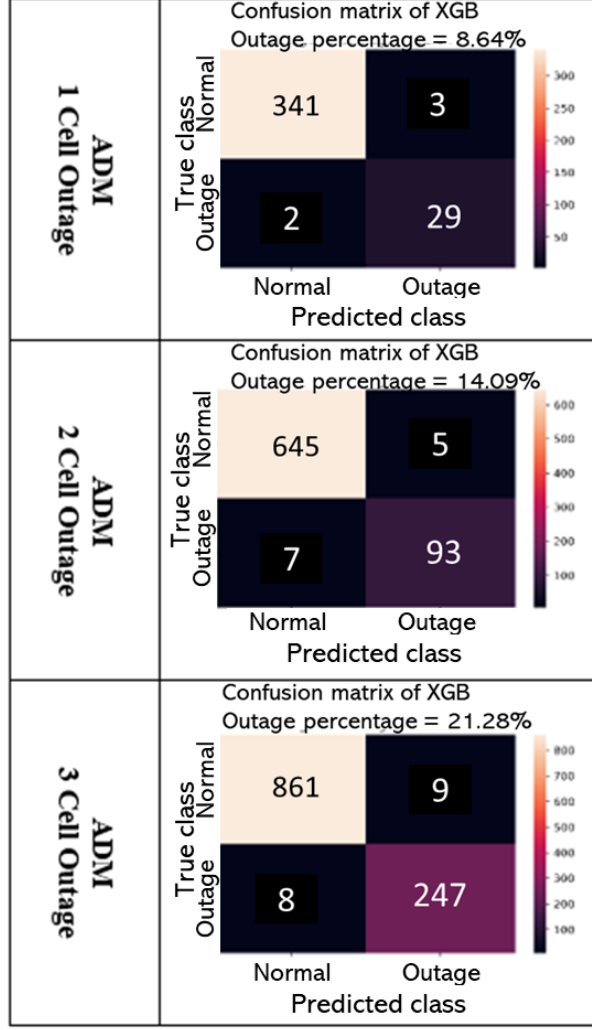


Fig. 5.7: Confusion matrices of XGBoost for the selected combination of MDT data density, and outage severity.

5.5.2 Anomaly Detection Module with XGBoost

For XGBoost training and testing, we generate labeled synthetic MDT data using SyntheticNet [243]. For real implementation of the proposed solution either a simulator or a digit twin of the network combined with the real MDT reports can be used. The data having features (RSRP and RSRQ from the six strongest neighboring BSs) is fed into the XGBoost with the goal of developing an optimal model that can distinguish the normal reports from the malicious MDT reports. The confusion matrix for different number of cell outages for XGBoost are shown in Figure 5.7. The outage percentage when only 1 cell is in outage is 8.64%, while it slightly increased to 14.9% and 21.28% in case of 2 and

TABLE 5.4. Performance comparison of the anomaly detection module for the Autoencoder and XGBoost.

MDT Data Size	Outage Severity					
	01-Cell-Outage		02-Cell-Outage		03-Cell-Outage	
	AEN	XGB	AEN	XGB	AEN	XGB
Low=2500	0.952	0.921	0.932	0.909	0.914	0.906
Med=5000	0.953	0.973	0.950	0.939	0.934	0.932
High=7500	0.987	0.969	0.962	0.951	0.971	0.966

3 cell outage, respectively.

Hence for module 1 (ADM) experiments, we have considered different combinations of MDT (normal and outage) dataset size. We considered the effectiveness of the outage detection model with the increase in the overall data size, and the percentage of outage instances in that data while the percentage of fake outages remains constant, i.e., 1% of the MDT reports size. More specifically, here we consider the low, medium, and high density of MDT data size. We have also considered different cell outage scenarios, i.e., 1, 2, and 3 cells-based outage severity out of 9 cells. To compare the performance of the outage detection module, AE is evaluated against the XGBoost classification method. We get the results for the AE and XGBoost models for the F1-score as shown in Table 5.4. It can be observed that for the same data size, there is a degradation in the AE and XGBoost performance, with an increase in outage severity. This is because, given the increase in the outage percentage, the chances of misclassification increase, specifically for the AE scheme which is designed to separate the anomalies. Hence the performance degradation is relatively less than the XGBoost scheme. It is worthnoting that despite the performance differences discussed above, overall, both schemes in ADM perform reasonably well for the anomaly detection, i.e., to separate the normal and outage instances.

5.6 Analysis and results for Malicious Reports Filtering Module

The next important task for the detection of malicious MDT attacks is to separate the fake outages from the real outages by exploiting Malicious Reports Filtering Module (MRFM).

We exploit user’s geographical locations, mapped with the corresponding MDT reports as inputs for training the local outlier factor (LOF) model [232–234]. The rationale behind the use of LOF and geographical positions stem on the fact that user reported real outages are usually geographically collocated and associated with the same BS. The LOF utilizes the density of the points in the neighborhood for the outlier detection, and it can be trained to exploit the geographical co-location to separate the real and fake outages as shown in Figure 5.8

- The LOF model is trained on the real outages data, with parameter settings of 15 neighbors, 0.15 contamination and Euclidean distance, to learn the frontier as shown in the figure. It can be observed that with these setting. For the evaluation of second module MRFM of the proposed framework that is designed to detect fake outages from the real outage MDT/CSI reports, we also defined three different cases of real and fake outage percentages and test the LOF scheme’s efficacy. We test the model performance when the fake outage percentage takes values from low to medium and high cases, and the real outage percentage remains constant. For this module, we consider 2-cell outage scenario, however, instead of a fixed fake outage percentage, here we changed the malicious outage rate from 5% to 90%.
- The results for this case are listed in Figure 5.8 The error rate, given as the ratio of the number of misclassified labels to the total number of labels in a particular class is given on the x-axis. Here, along with the MDT data, the geographical location of the MDT-generating UEs is exploited to separate the real outages from the malicious/fake outages. It is observed from these results that the PCA-assisted LOF-based separation in the MRFM learns the frontier of the real outages and is able to correctly filter the large percentage of fake outages based on the learned frontier.

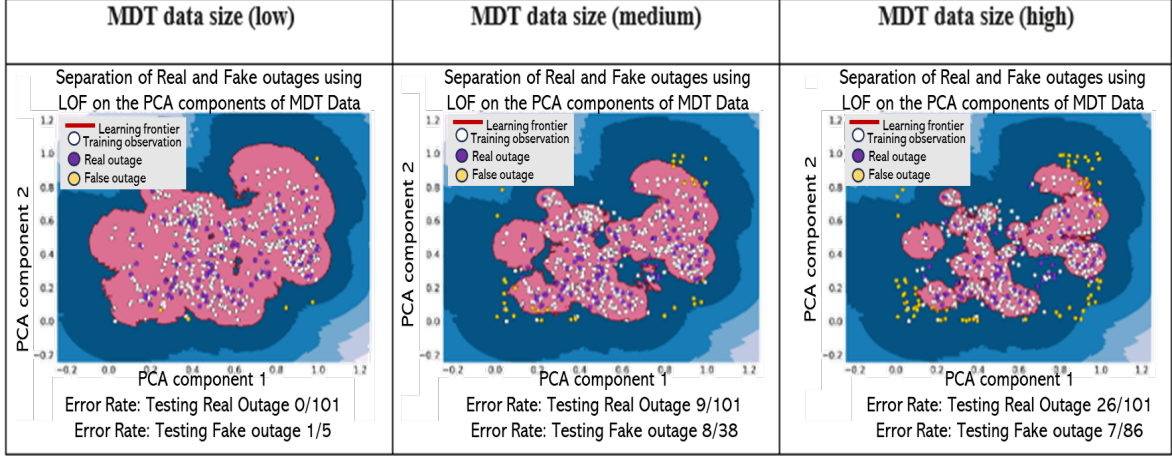


Fig. 5.8: Comparison of real and fake outage separation for MRFM using LOF.

The results verify that the proposed MRIF defense mechanism can detect adversarial attack by leveraging ML for outage detection and further identification of the malicious outages with high accuracy. Eventually, enabling the network to block and restrict these attackers from causing any sub-optimality in the network.

5.7 Open Research Challenges

In this chapter, we discuss the open research challenges faced due to the unavailability of representative IoT cellular networks data, that is crucial for the training of machine learning techniques. And also the difficulty of modeling the dynamic networks, which therefore leads to the inability to capture the time sensitive changes in the real time and makes to detect the adaptive attacks a challenging task.

For ML security model training, we rely on the anomalous data which is very rare to find. Thus, it is likely to train the ML model on a highly imbalance data as the attacks are sparse events. Training the model using such imbalance datasets can significantly impact the performance of the classifiers. Sparsity of the training data can impact the training of ML model negatively. Recent studies [248–251] show that this negatively affect can be offset to a varying extent by exploiting a range of data enrichment techniques that offer a trade- off between implementation complexity and their ability to recover missing data depending on the nature of the sparse data. To address the sparsity problem classic

interpolation techniques such as matrix factorization spline can be used [251]. Matrix factorization spline techniques is suitable method for sparsity issue when there is spatial or auto-correlation in the data. As propagation conditions in small cells are mostly line of sight, the shadowing remains correlated, and generally the shadowing standard deviation is small. This results in a low ranked matrix, therefore, for such setting matrix factorization spline technique becomes a promising approach for addressing training data sparsity as it jointly exploits matrix factorization theory and convex optimization as explained in [251].

Other techniques like Generative Adversarial Networks (GANs) can also be used to enrich sparse data when there are latent features in the data [249, 250]. Transfer learning techniques can also be implemented to deal with the sparse training data challenge [252]. Transfer learning is useful when the sparse data have no latent features. A comparative analysis of the pros and cons of various techniques for addressing data sparsity challenge in context of this framework is a separate extensive endeavor in itself and can be subject of a future extension of this study.

It represents the fact that even if we succeed to observe all the features of the network, for the training of the ML model, for its normal behavior to classify the anomalous behavior (outage, performance degradation), the system is time variant, and it keeps on evolving. Due to the non-stationary nature of the network features they keep changing over the time, and such trained model can be easily tampered by the attackers. To overcome this challenge the network should be trained on the large scale of spatiotemporal data [253].

5.8 Conclusion

In this chapter, we investigated the critical security vulnerability posed by adversarial attacks on Minimization of Drive Test (MDT) data, a foundational input for AI-based network automation in mobile and IoT networks. These attacks involve the deliberate manipulation of MDT reports by compromised user equipment, which can trigger the unintended activation or deactivation of self-organizing network (SON) or other zero-touch

automation functions. Our findings demonstrate that such manipulated inputs can lead to cascading failures, degrading network performance, misallocating resources, and undermining the reliability expected from next-generation cellular systems. To address this challenge, we proposed the Malicious MDT Reports Identification Framework (MRIF)—a robust, machine learning-based detection system capable of distinguishing between authentic and malicious MDT reports with high accuracy. Evaluated in the context of cell outage compensation (COC), one of the critical SON functions, MRIF successfully filtered adversarial data and preserved the integrity of automated network recovery processes. By incorporating a two-stage pipeline for anomaly detection and malicious report filtration, MRIF provides a proactive security layer for future cellular networks. Beyond its impact on network reliability, MRIF holds substantial importance for data-driven public health applications such as EpiTrack, where MDT data is leveraged for real-time mobility tracing and outbreak prediction. By ensuring the authenticity of mobility data at the network level, MRIF safeguards the reliability of such AI-powered epidemic forecasting frameworks. In conclusion, MRIF addresses a critical gap in securing mobile network intelligence by sustaining trust and stability in AI-integrated ecosystems.

CHAPTER 6

Conclusions and Future Work

6.1 Conclusion

The rapid evolution of mobile networks and the exponential rise in connected devices have led to a paradigm shift across domains such as smart healthcare, autonomous systems, and intelligent mobility. These interconnected digital ecosystems promise real-time situational awareness, predictive intelligence, and autonomous decision-making capabilities at unprecedented scale. However, as systems grow in complexity and interdependence, ensuring uninterrupted service quality, resilience, and trustworthiness becomes increasingly challenging due to several key roadblocks—including limited generalizability, delayed responsiveness, network vulnerabilities, and the lack of scalable intelligence.

To realize intelligent, proactive, and secure infrastructures, this dissertation identifies four fundamental barriers and proposes corresponding AI-driven frameworks to overcome them across individual, population, and network scales.

Existing health screening tools often rely on poorly balanced datasets, leading to limited generalization and unreliable predictions in real-world deployment. *SmartScreening*, the first contribution of this dissertation, addresses this challenge by proposing an end-to-end AI screening framework combining lightweight on-device classification with deep learning models on the cloud. It handles data noise, label scarcity, and demographic confounders by leveraging multi-modal acoustic features and semi-supervised learning, enabling scalable, fair, and resource-aware diagnostics for edge and cloud environments.

Traditional outbreak prediction models are reactive and heavily dependent on historical demographic or transportation data. In response, the second contribution, *EpiTrack*, introduces a novel AI-based framework that fuses real-time user mobility traces from mobile networks with smartphone-acquired respiratory signals to detect emerging hotspots be-

fore they escalate. Through synthetic mobility simulations, graph-based learning (GAT), and temporal modeling (LSTM, Positional Transformers), the framework supports early, location-specific interventions—pushing epidemic management from reactive containment to proactive mitigation.

Modern networks supporting smart systems lack foresight in detecting and responding to system-level degradations. The third contribution addresses this gap by introducing a stochastic reliability framework using Discrete Time Markov Chains (DTMC) to model the reliability state transitions of key network automation functions. This proactive monitoring framework predicts performance degradation and supports intelligent fault management—ensuring zero-touch recovery and enhancing quality of service (QoS) in next-generation 6G environments.

AI-driven systems are susceptible to adversarial attacks, particularly when reliant on user-reported mobile traces or mobility diagnostics. To safeguard against such threats, the final contribution of this dissertation presents *MRIF*, a multi-module security framework for detecting and filtering malicious data. By combining anomaly detection and malicious report filtering, MRIF effectively prevents poisoning of optimization and prediction pipelines—enhancing the integrity of digital intelligence systems.

6.2 Future Work

While this dissertation lays a comprehensive foundation across AI-driven diagnostics, epidemic forecasting, proactive fault prediction, and security in mobile networks, several pathways remain to elevate the scalability, adaptability, and trustworthiness of these systems. Building on the cough-based SmartScreening framework, future research will focus on designing multi-label and cross-domain learning models that generalize across diverse diseases and population segments. As smart device-based biomarker collection becomes increasingly pervasive, diagnostic frameworks must evolve beyond narrow detection to support holistic, individual-level health assessments capable of distinguishing

multiple co-occurring conditions in real-time. To extend the applicability of this work to the population level, the EpiTrack framework can be advanced by incorporating richer health context, such as additional biosignal modalities and behavioral dynamics. This progression would enable more nuanced modeling of outbreak transmission influenced by not just mobility, but real-world interactions, environment, and individual variability, laying the groundwork for adaptive public health interventions.

Enhancing network reliability is critical as ultra-reliable communication becomes the backbone for mission-critical applications across domains such as remote diagnostics, autonomous mobility, and emergency response. In these scenarios, even minimal downtime can lead to severe consequences, and thus reliability cannot be compromised. While the current DTMC-based fault prediction model offers a proactive foundation, future work must continue to evolve these frameworks. Incorporating reinforcement learning (RL) will enable the system to adaptively learn optimal reconfiguration and self-healing strategies in real time, further strengthening zero-touch automation and resilience in next-generation networks.

Lastly, to secure mission-critical applications such as real-time diagnostics and epidemic forecasting, robust cybersecurity mechanisms are indispensable. The MRIF security framework offers a foundation for safeguarding AI-enabled network operations from adversarial data attacks. Future extensions of this framework can explore federated learning and edge-aware detection strategies to enable privacy-preserving, distributed anomaly detection. Moreover, enhancing detection granularity using graph-based spatio-temporal correlations and ensemble machine learning techniques will fortify SON functions and reinforce the trustworthiness of AI-driven mobility intelligence, ensuring resilience against evolving threats.

To cohesively evaluate these multi-domain innovations under realistic conditions, a digital twin environment will be established to simulate outbreak dynamics, health diagnostics, and mobile network operations. This twin-based layer will serve as a testbed for refining learning models, validating control strategies, and ensuring zero-downtime deployment

across interconnected systems. By fusing generalizable AI, predictive modeling, and digital twin validation, this approach paves the way for the next generation of autonomous, resilient, and secure smart infrastructures. In summary, this dissertation lays the groundwork for building scalable, secure, and autonomous AI systems that support early detection, network resilience, and epidemic preparedness paving the way for intelligent digital infrastructures of the future.

Bibliography

- [1] Y. Zang, F. Zhang, C.-a. Di, and D. Zhu, “Advances of flexible pressure sensors toward Artificial Intelligence and health care applications,” *Materials Horizons*, vol. 2, no. 2, pp. 140–156, 2015.
- [2] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, Y. Wang, Q. Dong, H. Shen, and Y. Wang, “Artificial Intelligence in healthcare: past, present and future,” *Stroke and vascular neurology*, vol. 2, no. 4, pp. 230–243, 2017.
- [3] K. T. Chui, W. Alhalabi, S. S. H. Pang, P. O. d. Pablos, R. W. Liu, and M. Zhao, “Disease diagnosis in smart healthcare: Innovation, technologies and applications,” *Sustainability*, vol. 9, no. 12, p. 2309, 2017.
- [4] E. D. Esplin, L. Oei, and M. P. Snyder, “Personalized sequencing and the future of medicine: discovery, diagnosis and defeat of disease,” *Pharmacogenomics*, vol. 15, no. 14, pp. 1771–1790, 2014.
- [5] T. Davenport and R. Kalakota, “The potential for Artificial Intelligence in healthcare,” *Future healthcare journal*, vol. 6, no. 2, p. 94, 2019.
- [6] H. Liang, B. Y. Tsui, H. Ni, C. C. Valentim, S. L. Baxter, G. Liu, W. Cai, D. S. Kermany, X. Sun, J. Chen *et al.*, “Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence,” *Nature medicine*, vol. 25, no. 3, pp. 433–438, 2019.
- [7] M. A. Elaziz, K. M. Hosny, A. Salah, M. M. Darwish, S. Lu, and A. T. Sahlol, “New machine learning method for image-based diagnosis of covid-19,” *Plos one*, vol. 15, no. 6, p. e0235187, 2020.
- [8] C. Martin-Isla, V. M. Campello, C. Izquierdo, Z. Raisi-Estabragh, B. Baeßler, S. E. Petersen, and K. Lekadir, “Image-based cardiac diagnosis with machine learning: a review,” *Frontiers in cardiovascular medicine*, vol. 7, p. 1, 2020.
- [9] R. Palaniappan, K. Sundaraj, and N. U. Ahamed, “Machine learning in lung sound analysis: a systematic review,” *Biocybernetics and Biomedical Engineering*, vol. 33, no. 3, pp. 129–135, 2013.
- [10] R. Palaniappan, K. Sundaraj, and S. Sundaraj, “Artificial Intelligence techniques used in respiratory sound analysis—a systematic review,” *Biomedical Engineering/Biomedizinische Technik*, vol. 59, no. 1, pp. 7–18, 2014.

- [11] G. Chambres, P. Hanna, and M. Desainte-Catherine, “Automatic detection of patient with respiratory diseases using lung sound analysis,” in *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*. IEEE, 2018, pp. 1–6.
- [12] R. Folland, E. Hines, R. Dutta, P. Boilot, and D. Morgan, “Comparison of neural network predictors in the classification of tracheal–bronchial breath sounds by respiratory auscultation,” *Artificial Intelligence in medicine*, vol. 31, no. 3, pp. 211–220, 2004.
- [13] J. Sola-Soler, R. Jane, J. A. Fiz, and J. Morera, “Automatic classification of subjects with and without sleep apnea through snoring analysis,” in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2007, pp. 6093–6096.
- [14] I. Song, “Diagnosis of pneumonia from sounds collected using low cost cell phones,” in *2015 International joint conference on neural networks (IJCNN)*. IEEE, 2015, pp. 1–8.
- [15] P. Bokov, B. Mahut, P. Flaud, and C. Delclaux, “Wheezing recognition algorithm using recordings of respiratory sounds at the mouth in a pediatric population,” *Computers in biology and medicine*, vol. 70, pp. 40–50, 2016.
- [16] D. Perna and A. Tagarelli, “Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks,” in *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2019, pp. 50–55.
- [17] R. S. Irwin, C. L. French, A. B. Chang, K. W. Altman, T. M. Adams, E. Azoulay, A. F. Barker, S. S. Birring, F. Blackhall, D. C. Bolser *et al.*, “Classification of cough as a symptom in adults and management algorithms: CHEST guideline and expert panel report,” *Chest*, vol. 153, no. 1, pp. 196–209, 2018.
- [18] R. Palaniappan, K. Sundaraj, N. U. Ahamed, A. Arjunan, and S. Sundaraj, “Computer-based respiratory sound analysis: a systematic review,” *IETE Technical Review*, vol. 30, no. 3, pp. 248–256, 2013.
- [19] N. Das, M. Topalovic, and W. Janssens, “Artificial Intelligence in diagnosis of obstructive lung disease: current status and future potential,” *Current opinion in pulmonary medicine*, vol. 24, no. 2, pp. 117–123, 2018.
- [20] C. Qin, D. Yao, Y. Shi, and Z. Song, “Computer-aided detection in chest radiography based on artificial intelligence: a survey,” *Biomedical engineering online*, vol. 17, no. 1, p. 113, 2018.

- [21] A. Kumar, K. Abhishek, M. R. Ghalib, P. Nerurkar, K. Shah, M. Chandane, S. Bhirud, D. Patel, and Y. Busnel, "Towards cough sound analysis using the internet of things and deep learning for pulmonary disease prediction," *Transactions on emerging telecommunications technologies*, p. e4184, 2020.
- [22] A. Imran, I. Posokhova, H. N. Qureshi, U. Masood, S. Riaz, K. Ali, C. N. John, I. Hussain, and M. Nabeel, "AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app," *Informatics in Medicine Unlocked*, vol. 20, p. 100378, 2020.
- [23] G. Pinkas, Y. Karny, A. Malachi, G. Barkai, G. Bachar, and V. Aharonson, "SARS-CoV-2 detection from voice," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 268–274, 2020.
- [24] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3474–3484.
- [25] J. Laguarda, F. Hueto, and B. Subirana, "COVID-19 Artificial Intelligence Diagnosis using only Cough Recordings," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 275–281, 2020.
- [26] G.-T. Lee, H. Nam, S.-H. Kim, S.-M. Choi, Y. Kim, and Y.-H. Park, "Deep learning based cough detection camera using enhanced features," *arXiv preprint arXiv:2107.13260*, 2021.
- [27] M. E. Chowdhury, N. Ibtehaz, T. Rahman, Y. M. S. Mekki, Y. Qibalwey, S. Mahmud, M. Ezeddin, S. Zughaier, and S. A. S. Al-Maadeed, "QUCoughScope: An Artificially Intelligent Mobile Application to Detect Asymptomatic COVID-19 Patients using Cough and Breathing Sounds," *arXiv preprint arXiv:2103.12063*, 2021.
- [28] M. Pahar, M. Klopper, R. Warren, and T. Niesler, "COVID-19 Cough Classification using Machine Learning and Global Smartphone Recordings," *Computers in Biology and Medicine*, p. 104572, 2021.
- [29] Y. E. Erdoğan and A. Narin, "COVID-19 detection with traditional and deep features on cough acoustic signals," *Computers in Biology and Medicine*, vol. 136, p. 104765, 2021.
- [30] S. Rao, V. Narayanaswamy, M. Esposito, J. Thiagarajan, and A. Spanias, "Deep

- Learning with hyper-parameter tuning for COVID-19 Cough Detection,” in *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*. IEEE, 2021, pp. 1–5.
- [31] M. You, Z. Liu, C. Chen, J. Liu, X.-H. Xu, and Z.-M. Qiu, “Cough detection by ensembling multiple frequency subband features,” *Biomedical Signal Processing and Control*, vol. 33, pp. 132–140, 2017.
 - [32] T. H. Huynh, V. A. Tran, and H. D. Tran, “Semi-supervised tree support vector machine for online cough recognition,” in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.
 - [33] S.-H. Shin, T. Hashimoto, and S. Hatano, “Automatic detection system for cough sounds as a symptom of abnormal health condition,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 4, pp. 486–493, 2008.
 - [34] C. Pham, “MobiCough: real-time cough detection and monitoring using low-cost mobile devices,” in *Asian Conference on Intelligent Information and Database Systems*. Springer, 2016, pp. 300–309.
 - [35] B. Ferdousi, S. F. Ahsanullah, K. Abdullah-Al-Mamun, and M. N. Huda, “Cough detection using speech analysis,” in *2015 18th International Conference on Computer and Information Technology (ICCIT)*. IEEE, 2015, pp. 60–64.
 - [36] A. Van Hirtum and D. Berckmans, “Automated recognition of spontaneous versus voluntary cough,” *Medical engineering & physics*, vol. 24, no. 7-8, pp. 541–545, 2002.
 - [37] H. I. Hee, B. Balamurali, A. Karunakaran, D. Herremans, O. H. Teoh, K. P. Lee, S. S. Teng, S. Lui, and J. M. Chen, “Development of Machine Learning for Asthmatic and Healthy Voluntary Cough Sounds: A Proof of Concept Study,” *Applied Sciences*, vol. 9, no. 14, p. 2833, 2019.
 - [38] M. You, H. Wang, Z. Liu, C. Chen, J. Liu, X.-H. Xu, and Z.-M. Qiu, “Novel feature extraction method for cough detection using nmf,” *IET Signal Processing*, vol. 11, no. 5, pp. 515–520, 2017.
 - [39] J. Amoh and K. Odame, “DeepCough: A deep convolutional neural network in a wearable cough detection system,” in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2015, pp. 1–4.

- [40] P. S. Heckerling, T. G. Tape, R. S. Wigton, K. K. Hisson, J. B. Leikin, J. P. Ornato, J. L. Cameron, and E. M. Racht, “Clinical prediction rule for pulmonary infiltrates,” *Annals of internal medicine*, vol. 113, no. 9, pp. 664–670, 1990.
- [41] J. Korpáš, J. Sadloňová, and M. Vrabec, “Analysis of the cough sound: an overview,” *Pulmonary pharmacology*, vol. 9, no. 5-6, pp. 261–268, 1996.
- [42] J. A. Smith, H. L. Ashurst, S. Jack, A. A. Woodcock, and J. E. Earis, “The description of cough sounds by healthcare professionals,” *Cough*, vol. 2, no. 1, pp. 1–9, 2006.
- [43] Y. Khan, U. Qamar, N. Yousaf, and A. Khan, “Machine learning techniques for heart disease datasets: a survey,” in *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*, 2019, pp. 27–35.
- [44] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati, “Classification of ecg signals using machine learning techniques: A survey,” in *2015 International Conference on Advances in Computer Engineering and Applications*. IEEE, 2015, pp. 714–721.
- [45] F. Temurtas, “A comparative study on thyroid disease diagnosis using neural networks,” *Expert Systems with Applications*, vol. 36, no. 1, pp. 944–949, 2009.
- [46] T. J. Wroge, Y. Özkanca, C. Demiroglu, D. Si, D. C. Atkins, and R. H. Ghomi, “Parkinson’s disease diagnosis using machine learning and voice,” in *2018 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, 2018, pp. 1–7.
- [47] M. Nilashi, N. Ahmadi, S. Samad, L. Shahmoradi, H. Ahmadi, O. Ibrahim, S. Asadi, R. Abdullah, R. A. Abumalloh, and E. Yadegaridehkordi, “Disease diagnosis using machine learning techniques: A review and classification,” *Journal of Soft Computing and Decision Support Systems*, vol. 7, no. 1, pp. 19–30, 2020.
- [48] R. S. Irwin, M. H. Baumann, D. C. Bolser, L.-P. Boulet, S. S. Braman, C. E. Brightling, K. K. Brown, B. J. Canning, A. B. Chang, P. V. Dicpinigaitis *et al.*, “Diagnosis and management of cough executive summary: ACCP evidence-based clinical practice guidelines,” *Chest*, vol. 129, no. 1, pp. 1S–23S, 2006.
- [49] N. E. Wiseman, “The diagnosis of foreign body aspiration in childhood,” *Journal of pediatric surgery*, vol. 19, no. 5, pp. 531–535, 1984.
- [50] D. Singh, A. Agusti, A. Anzueto, P. J. Barnes, J. Bourbeau, B. R. Celli, G. J.

- Criner, P. Frith, D. M. Halpin, M. Han *et al.*, “Global strategy for the diagnosis, management, and prevention of chronic obstructive lung disease: the GOLD science committee report 2019,” *European Respiratory Journal*, vol. 53, no. 5, 2019.
- [51] C. Infante, D. Chamberlain, R. Fletcher, Y. Thorat, and R. Kodgule, “Use of cough sounds for diagnosis and screening of pulmonary disease,” in *2017 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2017, pp. 1–10.
- [52] I. D. Miranda, A. H. Diacon, and T. R. Niesler, “A Comparative Study of Features for Acoustic Cough Detection Using Deep Architectures,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 2601–2605.
- [53] M. Soliński, M. Lepek, and Ł. Kołowski, “Automatic cough detection based on airflow signals for portable spirometry system,” *Informatics in Medicine Unlocked*, p. 100313, 2020.
- [54] Y. Roh, G. Heo, and S. E. Whang, “A survey on data collection for machine learning: a big data-ai integration perspective,” *IEEE Transactions on Knowledge and Data Engineering*, 2019.
- [55] T. Drugman, J. Urbain, N. Bauwens, R. Chessini, A.-S. Aubriot, P. Lebecque, and T. Dutoit, “Audio and contact microphones for cough detection,” *arXiv preprint arXiv:2005.05313*, 2020.
- [56] W. Wei, J. Wang, J. Ma, N. Cheng, and J. Xiao, “A Real-time Robot-based Auxiliary System for Risk Evaluation of COVID-19 Infection,” *arXiv preprint arXiv:2008.07695*, 2020.
- [57] M. M. Baig, H. GholamHosseini, and M. J. Connolly, “Mobile healthcare applications: system design review, critical issues and challenges,” *Australasian physical & engineering sciences in medicine*, vol. 38, no. 1, pp. 23–38, 2015.
- [58] S. Matos, S. S. Birring, I. D. Pavord, and H. Evans, “Detection of cough signals in continuous audio recordings using hidden Markov models,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 1078–1083, 2006.
- [59] S. Matos, S. S. Birring, I. D. Pavord, and D. H. Evans, “An automated system for 24-h monitoring of cough frequency: the leicester cough monitor,” *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 8, pp. 1472–1479, 2007.

- [60] J. Martinek, M. Tatar, and M. Javorka, “Distinction between voluntary cough sound and speech in volunteers by spectral and complexity analysis,” *J Physiol Pharmacol*, vol. 59, no. 6, pp. 433–440, 2008.
- [61] B. H. Tracey, G. Comina, S. Larson, M. Bravard, J. W. López, and R. H. Gilman, “Cough detection algorithm for monitoring patient recovery from pulmonary tuberculosis,” in *2011 Annual international conference of the IEEE engineering in medicine and biology society*. IEEE, 2011, pp. 6017–6020.
- [62] T. Drugman, J. Urbain, N. Bauwens, R. Chessini, A.-S. Aubriot, P. Lebecque, and T. Dutoit, “Audio and contact microphones for cough detection,” in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.
- [63] J.-M. Liu, M. You, G.-Z. Li, Z. Wang, X. Xu, Z. Qiu, W. Xie, C. An, and S. Chen, “Cough signal recognition with gammatone cepstral coefficients,” in *2013 IEEE China Summit and International Conference on Signal and Information Processing*. IEEE, 2013, pp. 160–164.
- [64] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu, and Z. Qiu, “Cough detection using deep neural networks,” in *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2014, pp. 560–563.
- [65] T. Drugman, “Using mutual information in supervised temporal event detection: Application to cough detection,” *Biomedical Signal Processing and Control*, vol. 10, pp. 50–57, 2014.
- [66] M. Sterling, H. Rhee, and M. Bocko, “Automated cough assessment on a mobile platform,” *Journal of medical engineering*, vol. 2014, 2014.
- [67] H.-H. Wang, J.-M. Liu, M. You, and G.-Z. Li, “Audio signals encoding for cough classification using convolutional neural networks: A comparative study,” in *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2015, pp. 442–445.
- [68] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu, and Z. Qiu, “Cough event classification by pretrained deep neural network,” *BMC medical informatics and decision making*, vol. 15, no. S4, p. S2, 2015.
- [69] P. Klco, M. Kollarik, and M. Tatar, “Novel computer algorithm for cough monitoring based on octonions,” *Respiratory physiology & neurobiology*, vol. 257, pp. 36–41, 2018.

- [70] S. Khomsay, R. Vanijjirattikhan, and J. Suwatthikul, “Cough detection using PCA and Deep Learning,” in *2019 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2019, pp. 101–106.
- [71] M. Pahar, M. Klopfer, B. Reeve, G. Theron, R. Warren, and T. Niesler, “Automatic Cough Classification for Tuberculosis Screening in a Real-World Environment,” *arXiv preprint arXiv:2103.13300*, 2021.
- [72] P. Moradshahi, H. Chatrzarrin, and R. Goubiran, “Improving the performance of cough sound discriminator in reverberant environments using microphone array,” in *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*. IEEE, 2012, pp. 20–23.
- [73] E. C. Larson, T. Lee, S. Liu, M. Rosenfeld, and S. N. Patel, “Accurate and privacy preserving cough sensing using a low-cost microphone,” in *Proceedings of the 13th international conference on Ubiquitous computing*, 2011, pp. 375–384.
- [74] J. Monge-Alvarez, C. Hoyos-Barceló, P. Lesso, J. Escudero, K. Dahal, and P. Casaseca-de-la Higuera, “Effect of importance sampling on robust segmentation of audio-cough events in noisy environments,” in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2016, pp. 3740–3744.
- [75] C. Hoyos-Barcelo, J. Monge-Alvarez, M. Z. Shakir, J.-M. Alcaraz-Calero, and P. Casaseca-de La-Higuera, “Efficient k-NN implementation for real-time detection of cough events in smartphones,” *IEEE journal of biomedical and health informatics*, vol. 22, no. 5, pp. 1662–1671, 2017.
- [76] C. Hoyos-Barcelo, J. Monge-Álvarez, Z. Pervez, L. M. San-José-Revuelta, and P. Casaseca-de-la Higuera, “Efficient computation of image moments for robust cough detection using smartphones,” *Computers in biology and medicine*, vol. 100, pp. 176–185, 2018.
- [77] R. V. Sharan, U. R. Abeyratne, V. R. Swarnkar, S. Claxton, C. Hukins, and P. Porter, “Predicting spirometry readings using cough sound features and regression,” *Physiological measurement*, vol. 39, no. 9, p. 095001, 2018.
- [78] J. Monge-Álvarez, C. Hoyos-Barceló, P. Lesso, and P. Casaseca-de-la Higuera, “Robust detection of audio-cough events using local hu moments,” *IEEE journal of biomedical and health informatics*, vol. 23, no. 1, pp. 184–196, 2018.
- [79] J. Monge-Alvarez, C. Hoyos-Barceló, K. Dahal, and P. Casaseca-de-la Higuera,

- “Audio-cough event detection based on moment theory,” *Applied Acoustics*, vol. 135, pp. 124–135, 2018.
- [80] J. Monge-Álvarez, C. Hoyos-Barceló, L. M. San-José-Revuelta, and P. Casaseca-de-la Higuera, “A machine hearing system for robust cough detection based on a high-level representation of band-specific audio features,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 8, pp. 2319–2330, 2018.
 - [81] A. Windmon, M. Minakshi, P. Bharti, S. Chellappan, M. Johansson, B. A. Jenkins, and P. R. Athilingam, “TussisWatch: A Smart-Phone System to Identify Cough Episodes as Early Symptoms of Chronic Obstructive Pulmonary Disease and Congestive Heart Failure,” *IEEE journal of biomedical and health informatics*, vol. 23, no. 4, pp. 1566–1573, 2018.
 - [82] R. V. Sharan, U. R. Abeyratne, V. R. Swarnkar, and P. Porter, “Automatic croup diagnosis using cough sound recognition,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 2, pp. 485–495, 2018.
 - [83] M. J. Rahman, E. Nemati, M. Rahman, K. Vatanparvar, V. Nathan, and J. Kuang, “Efficient Online Cough Detection with a Minimal Feature Set Using Smartphones for Automated Assessment of Pulmonary Patients.”
 - [84] B. L. Y. Agbley, J. Li, A. Haq, B. Cobbinah, D. Kulevome, P. A. Agbefu, and B. Eleeza, “Wavelet-Based Cough Signal Decomposition for Multimodal Classification,” in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*. IEEE, 2020, pp. 5–9.
 - [85] S. R. Danda and B. Chen, “Toward Mitigating Spreading of Coronavirus via Mobile Devices,” *IEEE Internet of Things Magazine*, vol. 3, no. 3, pp. 12–16, 2020.
 - [86] B. Balamurali, H. I. Hee, O. Teoh, K. Lee, S. Kapoor, D. Herremans, and J.-M. Chen, “Asthmatic versus healthy child classification based on cough and vocalised/A:/sounds,” *The Journal of the Acoustical Society of America*, vol. 148, no. 3, p. EL253, 2020.
 - [87] A. A. Abaza, J. B. Day, J. S. Reynolds, A. M. Mahmoud, W. T. Goldsmith, W. G. McKinney, E. L. Petsonk, and D. G. Frazer, “Classification of voluntary cough sound and airflow patterns for detecting abnormal pulmonary function,” *Cough*, vol. 5, no. 1, p. 8, 2009.
 - [88] K. Kosasih, U. R. Abeyratne, and V. Swarnkar, “High frequency analysis of cough sounds in pediatric patients with respiratory diseases,” in *2012 Annual International*

Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 2012, pp. 5654–5657.

- [89] U. R. Abeyratne, V. Swarnkar, A. Setyati, and R. Triasih, “Cough sound analysis can rapidly diagnose childhood pneumonia,” *Annals of biomedical engineering*, vol. 41, no. 11, pp. 2448–2462, 2013.
- [90] K. Kosasih, U. R. Abeyratne, V. Swarnkar, and R. Triasih, “Wavelet augmented cough analysis for rapid childhood pneumonia diagnosis,” *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 4, pp. 1185–1194, 2014.
- [91] L. Di Perna, G. Spina, S. Thackray-Nocera, M. G. Crooks, A. H. Morice, P. Soda, and A. C. den Brinker, “An automated and unobtrusive system for cough detection,” in *2017 IEEE Life Sciences Conference (LSC)*. IEEE, 2017, pp. 190–193.
- [92] T. Drugman, J. Urbain, N. Bauwens, R. Chessini, C. Valderrama, P. Lebecque, and T. Dutoit, “Objective study of sensor relevance for automatic cough detection,” *IEEE journal of biomedical and health informatics*, vol. 17, no. 3, pp. 699–707, 2013.
- [93] V. Swarnkar, U. R. Abeyratne, Y. A. Amrulloh, and A. Chang, “Automated algorithm for Wet/Dry cough sounds classification,” in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012, pp. 3147–3150.
- [94] V. Swarnkar, U. R. Abeyratne, A. B. Chang, Y. A. Amrulloh, A. Setyati, and R. Triasih, “Automatic identification of wet and dry cough in pediatric patients with respiratory diseases,” *Annals of biomedical engineering*, vol. 41, no. 5, pp. 1016–1028, 2013.
- [95] V. Swarnkar, U. R. Abeyratne, Y. Amrulloh, C. Hukins, R. Triasih, and A. Setyati, “Neural network based algorithm for automatic identification of cough sounds,” in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2013, pp. 1764–1767.
- [96] Y. A. Amrulloh, U. R. Abeyratne, V. Swarnkar, R. Triasih, and A. Setyati, “Automatic cough segmentation from non-contact sound recordings in pediatric wards,” *Biomedical Signal Processing and Control*, vol. 21, pp. 126–136, 2015.
- [97] P. Kadambi, A. Mohanty, H. Ren, J. Smith, K. McGuinness, K. Holt, A. Furtwaengler, R. Slepetysh, Z. Yang, J.-s. Seo *et al.*, “Towards a Wearable Cough Detector Based on Neural Networks,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2161–2165.

- [98] K. A. Nguyen and Z. Luo, “Cover your cough: detection of respiratory events with confidence using a smartwatch,” in *Conformal and Probabilistic Prediction and Applications*. PMLR, 2018, pp. 114–131.
- [99] J. Schröder, J. Anemiiller, and S. Goetze, “Classification of human cough signals using spectro-temporal gabor filterbank features,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6455–6459.
- [100] “The automatic recognition and counting of cough, author=Barry, Samantha J and Dane, Adrie D and Morice, Alyn H and Walmsley, Anthony D,” *Cough*, vol. 2, no. 1, p. 8, 2006.
- [101] E. L. Chuma and Y. Iano, “A Movement Detection System Using Continuous-Wave Doppler Radar Sensor and Convolutional Neural Network to Detect Cough and Other Gestures,” *IEEE Sensors Journal*, 2020.
- [102] D. Parker, J. Picone, A. Harati, S. Lu, M. H. Jenkyns, and P. M. Polgreen, “Detecting paroxysmal coughing from pertussis cases using voice recognition technology,” *PloS one*, vol. 8, no. 12, 2013.
- [103] B. Rocha, L. Mendes, R. Couceiro, J. Henriques, P. Carvalho, and R. P. Paiva, “Detection of explosive cough events in audio recordings by internal sound analysis,” in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017, pp. 2761–2764.
- [104] L. Orlandic, T. Teijeiro, and D. Atienza, “The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms,” *Scientific Data*, vol. 8, no. 1, pp. 1–10, 2021.
- [105] N. Sharma, P. Krishnan, R. Kumar, S. Ramoji, S. R. Chetupalli, P. K. Ghosh, S. Ganapathy *et al.*, “Coswara—a database of breathing, cough, and voice sounds for covid-19 diagnosis,” *arXiv preprint arXiv:2005.10548*, 2020.
- [106] M. Cohen-McFarlane, R. Goubran, and F. Knoefel, “Novel coronavirus cough database: Nococoda,” *IEEE Access*, vol. 8, pp. 154 087–154 094, 2020.
- [107] “virufy Kernel Description,” <https://github.com/virufy/covid>, accessed: 2010-09-30.
- [108] Y. Amrulloh, U. Abeyratne, V. Swarnkar, and R. Triasih, “Cough sound analysis

- for pneumonia and asthma classification in pediatric population,” in *2015 6th International Conference on Intelligent Systems, Modelling and Simulation*. IEEE, 2015, pp. 127–131.
- [109] A. Keerio, B. K. Mitra, P. Birch, R. Young, and C. Chatwin, “On preprocessing of speech signals,” *International Journal of Signal Processing*, vol. 5, no. 3, pp. 216–222, 2009.
- [110] “IBM kernel description,” <https://developer.ibm.com/technologies/data-science/articles/data-preprocessing-in-detail/>, accessed: 2010-09-30.
- [111] M. B. Alsabek, I. Shahin, and A. Hassan, “Studying the similarity of covid-19 sounds based on correlation analysis of mfcc,” in *2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)*. IEEE, 2020, pp. 1–5.
- [112] V. P. Singh, J. Rohith, and V. K. Mittal, “Preliminary analysis of cough sounds,” in *2015 Annual IEEE India Conference (INDICON)*. IEEE, 2015, pp. 1–6.
- [113] H. Chatzarrin, A. Arcelus, R. Goubran, and F. Knoefel, “Feature extraction for the differentiation of dry and wet cough sounds,” in *2011 IEEE international symposium on medical measurements and applications*. IEEE, 2011, pp. 162–166.
- [114] W. Thorpe, M. Kurver, G. King, and C. Salome, “Acoustic analysis of cough,” in *The Seventh Australian and New Zealand Intelligent Information Systems Conference, 2001*. IEEE, 2001, pp. 391–394.
- [115] R. V. Sharan, U. R. Abeyratne, V. R. Swarnkar, and P. Porter, “Cough sound analysis for diagnosing croup in pediatric patients using biologically inspired features,” in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017, pp. 4578–4581.
- [116] J. Andreu-Perez, H. Pérez-Espinosa, E. Timonet, M. Kiani, M. I. Giron-Perez, A. B. Benitez-Trinidad, D. Jarchi, A. Rosales, N. Gkatzoulis, O. F. Reyes-Galaviz *et al.*, “A Generic Deep Learning Based Cough Analysis System from Clinically Validated Samples for Point-of-Need Covid-19 Test and Severity Levels,” *IEEE Transactions on Services Computing*, 2021.
- [117] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, “Mel-generalized cepstral analysis-a unified approach to speech spectral estimation,” in *Third International Conference on Spoken Language Processing*, 1994.

- [118] R. J. Mammone, X. Zhang, and R. P. Ramachandran, “Robust speaker recognition: A feature-based approach,” *IEEE signal processing magazine*, vol. 13, no. 5, p. 58, 1996.
- [119] B. W. Schuller, D. M. Schuller, K. Qian, J. Liu, H. Zheng, and X. Li, “COVID-19 and Computer Audition: An Overview on What Speech & Sound Analysis Could Contribute in the SARS-CoV-2 Corona Crisis,” *arXiv preprint arXiv:2003.11117*, 2020.
- [120] D. Moffat, D. Ronan, and J. D. Reiss, “An evaluation of audio feature extraction toolboxes,” 2015.
- [121] L. Kvapilova, V. Boza, P. Dubec, M. Majernik, J. Bogar, J. Jamison, J. C. Goldsack, D. J. Kimmel, and D. R. Karlin, “Continuous Sound Collection Using Smartphones and Machine Learning to Measure Cough,” *Digital Biomarkers*, vol. 3, no. 3, pp. 166–175, 2019.
- [122] G. Chaudhari, X. Jiang, A. Fakhry, A. Han, J. Xiao, S. Shen, and A. Khanzada, “Virufy: Global Applicability of Crowdsourced and Clinical Datasets for AI Detection of COVID-19 from Cough,” *arXiv preprint arXiv:2011.13320*, 2020.
- [123] “Coughtest Kernel Description,” <https://coughtest.online/>, accessed: 2010-09-30.
- [124] “coughdetect kernel description,” <https://coughdetect.com/sign-in>, accessed: 2010-09-30.
- [125] P. Mouawad, T. Dubnov, and S. Dubnov, “Robust Detection of COVID-19 in Cough Sounds,” *SN Computer Science*, vol. 2, no. 1, pp. 1–13, 2021.
- [126] G. C. Kagadis, C. Kloukinas, K. Moore, J. Philbin, P. Papadimitroulas, C. Alexakos, P. G. Nagy, D. Visvikis, and W. R. Hendee, “Cloud computing in medical imaging,” *Medical physics*, vol. 40, no. 7, 2013.
- [127] C. Pino and R. Di Salvo, “A survey of cloud computing architecture and applications in health,” in *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*. Atlantis Press, 2013, pp. 1649–1653.
- [128] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

- [129] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” *arXiv preprint arXiv:1904.08779*, 2019.
- [130] “1957-1958 Pandemic (H2N2 virus),” Accessed on: Aug. 16, 2021 [Online]. <https://www.cdc.gov/flu/pandemic-resources/1957-1958-pandemic.html>.
- [131] J. Antonovics, M. E. Hood, and C. H. Baker, “Was the 1918 flu avian in origin?” *Nature*, vol. 440, no. 7088, pp. E9–E9, 2006.
- [132] N. P. Johnson and J. Mueller, “Updating the accounts: global mortality of the 1918-1920” Spanish” influenza pandemic,” *Bulletin of the History of Medicine*, pp. 105–115, 2002.
- [133] S. Žižek, *Pandemic!: COVID-19 shakes the world*. John Wiley & Sons, 2020.
- [134] R. J. Barro, J. F. Ursúa, and J. Weng, “The coronavirus and the great influenza pandemic: Lessons from the “spanish flu” for the coronavirus’s potential effects on mortality and economic activity,” National Bureau of Economic Research, Tech. Rep., 2020.
- [135] “COVID-19 Map - Johns Hopkins Coronavirus Resource Center,” Accessed on: Aug. 16, 2021 [Online]. <https://coronavirus.jhu.edu/map.html>.
- [136] “Global economy could lose over \$4 trillion due to COVID-19 impact on tourism | UNCTAD,” Accessed on: Aug. 10, 2021 [Online]. <https://unctad.org/news/global-economy-could-lose-over-4-trillion-due-covid-19-impact-tourism>.
- [137] W.-K. Wu, J.-M. Liou, C.-C. Hsu, Y.-H. Lin, and M.-S. Wu, “Pandemic preparedness in Taiwan,” *Nature biotechnology*, vol. 38, no. 8, pp. 932–933, 2020.
- [138] C. D. Kelly-Cirino, J. Nkengasong, H. Kettler, I. Tongio, F. Gay-Andrieu, C. Escadafal, P. Piot, R. W. Peeling, R. Gadde, and C. Boehme, “Importance of diagnostics in epidemic and pandemic preparedness,” *BMJ global health*, vol. 4, no. Suppl 2, p. e001179, 2019.
- [139] W. N. Harrington, C. M. Kackos, and R. J. Webby, “The evolution and future of influenza pandemic preparedness,” *Experimental & Molecular Medicine*, vol. 53, no. 5, pp. 737–749, 2021.
- [140] E. C. Carbo, I. A. Sidorov, J. C. Zevenhoven-Dobbe, E. J. Snijder, E. C. Claas,

- J. F. Laros, A. C. Kroes, and J. J. de Vries, “Coronavirus discovery by metagenomic sequencing: a tool for pandemic preparedness,” *Journal of Clinical Virology*, vol. 131, p. 104594, 2020.
- [141] J. T. Wu and B. J. Cowling, “The use of mathematical models to inform influenza pandemic preparedness and response,” *Experimental Biology and Medicine*, vol. 236, no. 8, pp. 955–961, 2011.
- [142] H. Heesterbeek, R. M. Anderson, V. Andreasen, S. Bansal, D. De Angelis, C. Dye, K. T. Eames, W. J. Edmunds, S. D. Frost, S. Funk *et al.*, “Modeling infectious disease dynamics in the complex landscape of global health,” *Science*, vol. 347, no. 6227, 2015.
- [143] J.-P. Chretien, S. Riley, and D. B. George, “Mathematical modeling of the West Africa Ebola epidemic,” *Elife*, vol. 4, p. e09186, 2015.
- [144] G. Chowell, L. Sattenspiel, S. Bansal, and C. Viboud, “Mathematical models to characterize early epidemic growth: A review,” *Physics of life reviews*, vol. 18, pp. 66–97, 2016.
- [145] S. Feng, Z. Feng, C. Ling, C. Chang, and Z. Feng, “Prediction of the COVID-19 epidemic trends based on SEIR and AI models,” *PLOS ONE*, vol. 16, pp. 1–15, 01 2021. [Online]. Available: <https://doi.org/10.1371/journal.pone.0245101>
- [146] X.-X. Liu, S. J. Fong, N. Dey, R. G. Crespo, and E. Herrera-Viedma, “A new SEAIRD pandemic prediction model with clinical and epidemiological data analysis on COVID-19 outbreak,” *Applied Intelligence*, pp. 1–37, 2021.
- [147] Y. Fang, Y. Nie, and M. Penny, “Transmission dynamics of the covid-19 outbreak and effectiveness of government interventions: A data-driven analysis,” *Journal of medical virology*, vol. 92, no. 6, pp. 645–659, 2020.
- [148] J. Lou, H. Zheng, S. Zhao, L. Cao, E. L. Wong, Z. Chen, R. W. Chan, M. K. Chong, B. C. Zee, P. K. Chan *et al.*, “Quantifying the effect of government interventions and virus mutations on transmission advantage during covid-19 pandemic,” *Journal of Infection and Public Health*, vol. 15, no. 3, pp. 338–342, 2022.
- [149] B. Du, Z. Zhao, J. Zhao, L. Yu, L. Sun, and W. Lv, “Modelling the epidemic dynamics of covid-19 with consideration of human mobility,” *International Journal of Data Science and Analytics*, vol. 12, no. 4, pp. 369–382, 2021.

- [150] E. Hernández-Orallo and A. Armero-Martínez, “How human mobility models can help to deal with covid-19,” *Electronics*, vol. 10, no. 1, p. 33, 2020.
- [151] M. Li, S. Ma, and Z. Liu, “A novel method to detect the early warning signal of COVID-19 transmission,” *BMC Infectious Diseases*, vol. 22, no. 1, pp. 1–12, 2022.
- [152] Y. Karadayi, M. N. Aydin, and A. S. Öğrenci, “Unsupervised anomaly detection in multivariate spatio-temporal data using deep learning: early detection of covid-19 outbreak in italy,” *Ieee Access*, vol. 8, pp. 164 155–164 177, 2020.
- [153] H. Homayouni, I. Ray, S. Ghosh, S. Gondalia, and M. G. Kahn, “Anomaly detection in covid-19 time-series data,” *SN Computer Science*, vol. 2, no. 4, p. 279, 2021.
- [154] Z. Cui, M. Cai, Y. Xiao, Z. Zhu, M. Yang, and G. Chen, “Forecasting the transmission trends of respiratory infectious diseases with an exposure-risk-based model at the microscopic level,” *Environmental Research*, vol. 212, p. 113428, 2022.
- [155] Z. Cui, M. Cai, Y. Xiao, Z. Zhu, and G. Chen, “Influences of obstacle factors on the transmission trends of respiratory infectious diseases in indoor public places,” *Journal of Building Engineering*, vol. 64, p. 105706, 2023.
- [156] M. Tizzoni, P. Bajardi, A. Decuyper, G. Kon Kam King, C. M. Schneider, V. Blondel, Z. Smoreda, M. C. González, and V. Colizza, “On the use of human mobility proxies for modeling epidemics,” *PLoS computational biology*, vol. 10, no. 7, p. e1003716, 2014.
- [157] A. Schwabe, J. Persson, and S. Feuerriegel, “Predicting covid-19 spread from large-scale mobility data,” in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 3531–3539.
- [158] P. Li, X. Chen, C. Ma, C. Zhu, and W. Lu, “Risk assessment of covid-19 infection for subway commuters integrating dynamic changes in passenger numbers,” *Environmental Science and Pollution Research*, vol. 29, no. 49, pp. 74 715–74 724, 2022.
- [159] M.-C. Chang, R. Kahn, Y.-A. Li, C.-S. Lee, C. O. Buckee, and H.-H. Chang, “Variation in human mobility and its impact on the risk of future covid-19 outbreaks in taiwan,” *BMC public health*, vol. 21, no. 1, pp. 1–10, 2021.
- [160] C. Fan, R. Lee, Y. Yang, and A. Mostafavi, “Fine-grained data reveal segregated mobility networks and opportunities for local containment of covid-19,” *Scientific*

Reports, vol. 11, no. 1, p. 16895, 2021.

- [161] S. Zhu, A. Bukharin, L. Xie, K. Yamin, S. Yang, P. Keskinocak, and Y. Xie, “Early detection of covid-19 hotspots using spatio-temporal data,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 2, pp. 250–260, 2022.
- [162] T. Yabe, K. Tsubouchi, Y. Sekimoto, and S. V. Ukkusuri, “Early warning of covid-19 hotspots using human mobility and web search query data,” *Computers, Environment and Urban Systems*, vol. 92, p. 101747, 2022.
- [163] M. Raghavan, K. S. Sridharan, and Y. Mandayam Rangayyan, “Using epidemic simulators for monitoring an ongoing epidemic,” *Scientific Reports*, vol. 10, no. 1, p. 16571, 2020.
- [164] “Wearables - statistics facts,” Accessed on: Jan. 10, 2025 [Online]. <https://www.statista.com/topics/1556/wearable-technology/statisticChapter>.
- [165] “Global Wearable Technology Market,” Accessed on: Jan. 10, 2025 [Online]. <https://www.imarcgroup.com/wearable-technology-market-statistics?>
- [166] “Wearable Medical Devices Market,” Accessed on: Jan. 10, 2025 [Online]. <https://www.globenewswire.com/news-release/2024/10/15/2963602/0/en/Wearable-Medical-Devices-Market-to-Soar>.
- [167] N. Pica and N. M. Bouvier, “Environmental factors affecting the transmission of respiratory viruses,” *Current opinion in virology*, vol. 2, no. 1, pp. 90–95, 2012.
- [168] A. K. Weaver, J. R. Head, C. F. Gould, E. J. Carlton, and J. V. Remais, “Environmental factors influencing covid-19 incidence and severity,” *Annual review of public health*, vol. 43, pp. 271–291, 2022.
- [169] K. Hope, R. Boyd, S. Conaty, and P. Maywood, “Measles transmission in health care waiting rooms: implications for public health response,” *Western Pacific Surveillance and Response Journal: WPSAR*, vol. 3, no. 4, p. 33, 2012.
- [170] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, and M. B. Schwamborn, “A mobility scenario generation and analysis tool,” *Osnabruck, Germany*, 2013.
- [171] N. Aschenbruck and M. Schwamborn, “Synthetic map-based mobility traces for the performance evaluation in opportunistic networks,” in *Proceedings of the Second International Workshop on Mobile Opportunistic Networking*, 2010, pp. 143–146.

- [172] M. Schwamborn and N. Aschenbruck, “Introducing geographic restrictions to the slaw human mobility model,” in *2013 IEEE 21st international symposium on modelling, analysis and simulation of computer and telecommunication systems*. IEEE, 2013, pp. 264–272.
- [173] A. Sherstinsky, “Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network,” *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020.
- [174] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks,” *arXiv preprint arXiv:1710.10903*, 2017.
- [175] H. Mathur and T. Deepa, “A survey on advanced multiple access techniques for 5g and beyond wireless communications,” *Wireless Personal Communications*, vol. 118, no. 2, pp. 1775–1792, 2021.
- [176] H. Sanneck, “5G Network slicing management for challenged network scenarios,” in *Proceedings of the 12th Workshop on Challenged Networks*, 2017, pp. 19–20.
- [177] B. Rong, M. Dianati, L. Zhou, G. K. Karagiannidis, and C. Wang, “5G mmWave small cell networks: architecture, self-organization, and management,” *IEEE Wireless Communications*, vol. 25, no. 4, pp. 8–9, 2018.
- [178] M. Pérez, G. M. Perez, P. G. Giardina, G. Bernini, P. Neves, J. M. Alcaraz-Calero, Q. Wang, and K. Koutsopoulos, “Self-organizing capabilities in 5G networks: NFV & SDN coordination in a complex use case,” *Proceedings of EuCNC*, pp. 1–5, 2018.
- [179] Y. Koucheryavy, E. Lisovskaya, D. Moltchanov, R. Kovalchukov, and A. Samuylov, “Quantifying the millimeter wave new radio base stations density for network slicing with prescribed slas,” *Computer Communications*, vol. 174, pp. 13–27, 2021.
- [180] M. Kordestani, M. Saif, M. E. Orchard, R. Razavi-Far, and K. Khorasani, “Failure prognosis and applications—a survey of recent literature,” *IEEE transactions on reliability*, 2019.
- [181] A. Guidotti, A. Vanelli-Coralli, M. Conti, S. Andrenacci, S. Chatzinotas, N. Maturo, B. Evans, A. Awoseyila, A. Ugolini, T. Foggi *et al.*, “Architectures and key technical challenges for 5G systems incorporating satellites,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2624–2639, 2019.
- [182] X. Zhao, B. Liu, and Y. Liu, “Reliability modeling and analysis of load-sharing sys-

- tems with continuously degrading components,” *IEEE Transactions on Reliability*, vol. 67, no. 3, pp. 1096–1110, 2018.
- [183] M. Elhoseny and K. Shankar, “Reliable data transmission model for mobile ad hoc network using signcryption technique,” *IEEE Transactions on Reliability*, vol. 69, no. 3, pp. 1077–1086, 2019.
 - [184] Z. Yin, X. Ma, J. Zheng, Y. Zhou, L. N. Bairavasundaram, and S. Pasupathy, “An empirical study on configuration errors in commercial and open source systems,” in *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, 2011, pp. 159–172.
 - [185] S.-I. Yang, D. M. Frangopol, and L. C. Neves, “Service life prediction of structural systems using lifetime functions with emphasis on bridges,” *Reliability Engineering & System Safety*, vol. 86, no. 1, pp. 39–51, 2004.
 - [186] Q. Liu and Z. Zhang, “The analysis of coverage probability, ase and ee in heterogeneous ultra-dense networks with power control,” *Digital Communications and Networks*, vol. 6, no. 4, pp. 524–533, 2020.
 - [187] “Experimentation results and evaluation of achievements in terms of KPIs,” Accessed on: Oct. 16, 2021 [Online]. <https://vdocument.in/d53-experimentation-results-and-evaluation-of-5g-kpi-key-performance-indicator.html>.
 - [188] C.-P. Li, J. Jiang, W. Chen, T. Ji, and J. Smee, “5g ultra-reliable and low-latency systems design,” in *2017 European Conference on Networks and Communications (EuCNC)*. IEEE, 2017, pp. 1–5.
 - [189] “3GPP, TR 36.913: Requirements for Further Advancements for Evolved Universal Terrestrial Radio Access (E-UTRA),” Accessed on: Aug. 10, 2021 [Online]. www.3gpp.org, Tech. Rep., Mar. 2009.
 - [190] “3GPP, TS 32.500 version 10.1.0: Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Self-Organizing Networks (SON); Concepts and requirements,” Accessed on: Aug. 10, 2021 [Online]. www.3gpp.org, Tech. Rep., Mar. 2010.
 - [191] A. Imran, A. Zoha, and A. Abu-Dayya, “Challenges in 5G: how to empower SON with big data for enabling 5G,” *IEEE network*, vol. 28, no. 6, pp. 27–33, 2014.
 - [192] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, “A survey of self organisation

- in future cellular networks,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 336–361, 2012.
- [193] V. Räsänen and H. Tang, “Knowledge modeling for conflict detection in self-organized networks,” in *International Conference on Mobile Networks and Management*. Springer, 2011, pp. 107–119.
- [194] H. Y. Lateef, A. Imran, M. A. Imran, L. Giupponi, and M. Dohler, “Lte-advanced self-organizing network conflicts and coordination algorithms,” *IEEE Wireless Communications*, vol. 22, pp. 108–117, 2015.
- [195] H. Y. Lateef, A. Imran, and A. Abu-Dayya, “A framework for classification of self-organising network conflicts and coordination algorithms,” in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2013, pp. 2898–2903.
- [196] J. Xu, J. Yao, L. Wang, K. Wu, L. Chen, and W. Lou, “Revolution of self-organizing network for 5G mmWave small cell management: From reactive to proactive,” *IEEE Wireless Commun.*, vol. 25, pp. 66–73, 2018.
- [197] “Intent-based-networks,” Accessed on: Oct. 16, 2021 [Online]. <https://www.ericsson.com/en/ai/intent-based-networks>.
- [198] D. Rojas, F. Nazmetdinov, and A. Mitschele-Thiel, “Zero-touch coordination framework for Self-Organizing Functions in 5G,” in *2020 IEEE Wireless Commun. and Networking Conference (WCNC)*, 2020, pp. 1–8.
- [199] A. Asghar, H. Farooq, and A. Imran, “Concurrent optimization of coverage, capacity, and load balance in hetnets through soft and hard cell association parameters,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8781–8795, 2018.
- [200] J. Moysen, M. Garcia-Lozano, L. Giupponi, and S. Ruiz, “Conflict resolution in mobile networks: a self-coordination framework based on non-dominated solutions and machine learning for data analytics,” *IEEE Computational Intelligence Mag.*, vol. 13, no. 2, pp. 52–64, 2018.
- [201] T. Bag, S. Garg, D. F. P. Rojas, and A. Mitschele-Thiel, “Machine learning-based recommender systems to achieve self-coordination between son functions,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2131–2144, 2020.

- [202] O. Iacobaiea, B. Sayrac, S. B. Jemaa, and P. Bianchi, "SON Coordination for parameter conflict resolution: A reinforcement learning framework," in *2014 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2014, pp. 196–201.
- [203] C. Frenzel, B. Bauer, C. Schmelz, and H. Sanneck, "Objective-driven coordination in self-organizing networks," in *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2015, pp. 1453–1458.
- [204] A. Prasad, M. A. Uusitalo, Z. Li, and P. Lundén, "Reflection Environment Maps for Enhanced Reliability in 5G Self-Organizing Networks," in *2016 IEEE Globecom Workshops (GC Wkshps)*, 2016, pp. 1–6.
- [205] Č. Stefanović, "Industry 4.0 from 5G perspective: Use-cases, requirements, challenges and approaches," in *2018 11th CMI International Conference: Prospects and Challenges Towards Developing a Digital Economy within the EU*. IEEE, 2018, pp. 44–48.
- [206] A. Zoha, A. Saeed, H. Farooq, A. Rizwan, A. Imran, and M. A. Imran, "Leveraging intelligence from network cdr data for interference aware energy consumption minimization," *IEEE Transactions on Mobile Computing*, vol. 17, no. 7, pp. 1569–1582, 2017.
- [207] Y. Kumar, H. Farooq, and A. Imran, "Fault prediction and reliability analysis in a real cellular network," in *IEEE International Wireless Commun. and Mobile Computing (IWCMC)*, 2017, pp. 1090–1095.
- [208] A. Papoulis and H. Saunders, "Probability, random variables and stochastic processes," 1989.
- [209] S. Nikmanesh, M. Akbari, and R. Joda, "Proactive Self-Healing Analysis-Framework Based on Discrete-Time Markov Decision Process in 5G Network and Beyond," in *2018 9th International Symposium on Telecommunications (IST)*. IEEE, 2018, pp. 690–695.
- [210] H. Farooq, M. S. Parwez, and A. Imran, "Continuous time markov chain based reliability analysis for future cellular networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*, 2015, pp. 1–6.
- [211] S. M. A. Zaidi, M. Manalastas, H. Farooq, and A. Imran, "SyntheticNET: A 3GPP compliant simulator for AI enabled 5G and beyond," *IEEE Access*, vol. 8, pp.

82 938–82 950, 2020.

- [212] M. Z. Asghar, P. Nieminen, S. Hämäläinen, T. Ristaniemi, M. A. Imran, and T. Hämäläinen, “Towards proactive context-aware self-healing for 5G networks,” *Computer Networks*, vol. 128, pp. 5–13, 2017.
- [213] ““IoT: number of connected devices worldwide 2012-2025,” ,” Accessed on: March 10,2023 [Online]. Statista. <https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/>.
- [214] 3rd Generation Partnership Project (3GPP) Technical Specification Group Radio Access Network NR, “Radio resource control (RRC) protocol specification,” in *3GPP TS 38.331 version 16.0.0 Release 16*, 2020.
- [215] A. Scaloni, P. Cirella, M. Sgheiz, R. Diamanti, and D. Micheli, “Multipath and doppler characterization of an electromagnetic environment by massive mdt measurements from 3g and 4g mobile terminals,” *IEEE Access*, vol. 7, pp. 13 024–13 034, 2019.
- [216] “3GPP TS 37.320 V16.7.0 “3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Radio measurement collection for Minimization of Drive Tests (MDT); Overall description; Stage 2,” Accessed on: March 10,2023 [Online]. <https://www.3gpp.org/dynareport?code=37-series.html>.
- [217] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, “Internet of things (iot): Taxonomy of security attacks,” in *2016 3rd international conference on electronic design (ICED)*. IEEE, 2016, pp. 321–326.
- [218] J. Deogirikar and A. Vidhate, “Security attacks in iot: A survey,” in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, 2017, pp. 32–37.
- [219] J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, “Security and privacy for cloud-based IoT: Challenges,” *IEEE Communications Magazine*, vol. 55, no. 1, pp. 26–33, 2017.
- [220] H. A. Abdul-Ghani, D. Konstantas, and M. Mahyoub, “A comprehensive iot attacks survey based on a building-blocked reference model,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 355–373, 2018.
- [221] Z. Wang, D. Liu, Y. Sun, X. Pang, P. Sun, F. Lin, J. C. Lui, and K. Ren, “A

- survey on iot-enabled home automation systems: Attacks and defenses,” *IEEE Communications Surveys & Tutorials*, 2022.
- [222] K. Lounis and M. Zulkernine, “Attacks and defenses in short-range wireless technologies for iot,” *IEEE Access*, vol. 8, pp. 88 892–88 932, 2020.
 - [223] L. Xiao, X. Wan, X. Lu, Y. Zhang, and D. Wu, “IoT security techniques based on machine learning: How do IoT devices use AI to enhance security?” *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 41–49, 2018.
 - [224] X. Liang and Y. Kim, “A survey on security attacks and solutions in the iot network,” in *2021 IEEE 11th annual computing and communication workshop and conference (CCWC)*. IEEE, 2021, pp. 0853–0859.
 - [225] S. M. Tahsien, H. Karimipour, and P. Spachos, “Machine learning based solutions for security of internet of things (iot): A survey,” *Journal of Network and Computer Applications*, vol. 161, p. 102630, 2020.
 - [226] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, “A survey of self organisation in future cellular networks,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 336–361, 2012.
 - [227] S. Hämmäläinen, H. Sanneck, and C. Sartori, *LTE self-organising networks (SON): network management automation for operational efficiency*. John Wiley & Sons, 2012.
 - [228] ““Phones - OsmocomBB - Open Source Mobile Communications.”,” Accessed on: March 10,2023 [Online]. Statista. <https://osmocom.org/projects/baseband/wiki/Phones>.
 - [229] H. Y. Lateef, A. Imran, and A. Abu-Dayya, “A framework for classification of self-organising network conflicts and coordination algorithms,” in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2013, pp. 2898–2903.
 - [230] H. Y. Lateef, A. Imran, M. A. Imran, L. Giupponi, and M. Dohler, “LTE-advanced self-organizing network conflicts and coordination algorithms,” *IEEE Wireless Communications*, vol. 22, no. 3, pp. 108–117, 2015.
 - [231] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. A. Imran, and A. Abu-Dayya, “A cell outage management framework for dense heterogeneous networks,”

IEEE Transactions on Vehicular Technology, vol. 65, no. 4, pp. 2097–2113, 2015.

- [232] E. H. Budiarto, A. E. Permanasari, and S. Fauziati, “Unsupervised anomaly detection using k-means, local outlier factor and one class svm,” in *2019 5th international conference on science and technology (ICST)*, vol. 1. IEEE, 2019, pp. 1–5.
- [233] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, “High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning,” *Pattern Recognition*, vol. 58, pp. 121–134, 2016.
- [234] D. Pokrajac, A. Lazarevic, and L. J. Latecki, “Incremental local outlier detection for data streams,” in *2007 IEEE symposium on computational intelligence and data mining*. IEEE, 2007, pp. 504–515.
- [235] Y. Kang, M. Tan, D. Lin, and Z. Zhao, “Intrusion detection model based on autoencoder and xgboost,” in *Journal of Physics: Conference Series*, vol. 2171, no. 1. IOP Publishing, 2022, p. 012053.
- [236] X. Wang and X. Lu, “A host-based anomaly detection framework using XGBoost and LSTM for IoT devices,” *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–13, 2020.
- [237] A. Taghavirashidizadeh, M. Zavvar, M. Moghadaspour, M. Jafari, H. Garoosi, and M. H. Zavvar, “Anomaly detection in iot networks using hybrid method based on pca-xgboost,” in *2022 8th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIIS)*. IEEE, 2022, pp. 1–5.
- [238] Z. Chen, C. K. Yeo, B. S. Lee, and C. T. Lau, “Autoencoder-based network anomaly detection,” in *2018 Wireless telecommunications symposium (WTS)*. IEEE, 2018, pp. 1–5.
- [239] Z. Cheng, S. Wang, P. Zhang, S. Wang, X. Liu, and E. Zhu, “Improved autoencoder for unsupervised anomaly detection,” *International Journal of Intelligent Systems*, vol. 36, no. 12, pp. 7103–7125, 2021.
- [240] C. Zhou and R. C. Paffenroth, “Anomaly detection with robust deep autoencoders,” in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 665–674.
- [241] L. Xu, Y.-R. Yeh, Y.-J. Lee, and J. Li, “A hierarchical framework using approximated local outlier factor for efficient anomaly detection,” *Procedia Computer Sci-*

ence, vol. 19, pp. 1174–1181, 2013.

- [242] N. Paulauskas and A. F. Bagdonas, “Local outlier factor use for the network flow anomaly detection,” *Security and Communication Networks*, vol. 8, no. 18, pp. 4203–4212, 2015.
- [243] S. M. A. Zaidi, M. Manalastas, H. Farooq, and A. Imran, “Syntheticnet: A 3gpp compliant simulator for ai enabled 5g and beyond,” *IEEE Access*, vol. 8, pp. 82 938–82 950, 2020.
- [244] “Anomaly Detection using Autoencoders, ,” Accessed on: March 10,2023 [Online]. Statista. <https://towardsdatascience.com/anomaly-detection-using-autoencoders-5b032178a1ea>.
- [245] “Applying Anomaly Detection with Autoencoders to Fraud Detection, ,” Accessed on: March 10,2023 [Online]. <https://towardsdatascience.com/applying-anomaly-detection-with-autoencoders-to-fraud-detection-feaace6b5b09>.
- [246] “Anomaly Detection using autoencoders a walk through in python, ,” Accessed on: March 10,2023 [Online]. <https://www.analyticsvidhya.com/blog/2021/05/anomaly-detection-using-autoencoders-a-walk-through-in-python/>.
- [247] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano, and L. Benini, “Anomaly detection using autoencoders in high performance computing systems,” in *Proceedings of the AAAI Conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 9428–9433.
- [248] H. N. Qureshi, A. Imran, and A. Abu-Dayya, “Enhanced mdt-based performance estimation for ai driven optimization in future cellular networks,” *IEEE Access*, vol. 8, pp. 161 406–161 426, 2020.
- [249] B. Hughes, S. Bothe, H. Farooq, and A. Imran, “Generative adversarial learning for machine learning empowered self organizing 5g networks,” in *2019 international conference on computing, networking and communications (ICNC)*. IEEE, 2019, pp. 282–286.
- [250] J. Shodamola, H. Qureshi, U. Masood, and A. Imran, “Towards addressing the spatial sparsity of mdt reports to enable zero touch network automation,” in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.
- [251] H. N. Qureshi, U. Masood, M. Manalastas, S. M. A. Zaidi, H. Farooq, J. Forgeat,

- M. Bouton, S. Bothe, P. Karlsson, A. Rizwan *et al.*, “Towards addressing training data scarcity challenge in emerging radio access networks: A survey and framework,” *IEEE Communications Surveys & Tutorials*, 2023.
- [252] J. Liang, L. Li, and C. Zhao, “A transfer learning approach for compressed sensing in 6g-iot,” *IEEE Internet of Things Journal*, vol. 8, no. 20, pp. 15 276–15 283, 2021.
- [253] L. Yang and A. Shami, “A lightweight concept drift detection and adaptation framework for iot data streams,” *IEEE Internet of Things Magazine*, vol. 4, no. 2, pp. 96–101, 2021.

Funding Acknowledgments

The projects undertaken in this dissertation are based upon work supported by the National Science Foundation under Grant Numbers 1923669 and 1730650. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. For more inquiries about these works, please visit www.AI4networks.com.