République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université A.MIRA-BEJAIA

Faculté Sciences Exactes
Département Informatique

THÈSE

Présentée par

BELHADJ Asma

Pour l'obtention du grade de

DOCTEUR EN SCIENCES

Filière : Informatique

Option : Cloud Computing

Thème

Predictive Maintenance for Quality of Service in IoT Communications: 5G Use Case

Soutenue le : 11/09/2025

Devant le Jury composé de :

Nom et Prénom	Grade		
Mme BOUKREDERA Djamila	MCA	Univ. de Bejaia	Président
Mr OMAR Mawloud	Professeur	Univ. de Bretagne Sud	Rapporteur
Mr FARAH Zoubeyr	MCA	Univ. de Bejaia	Examinateur
Mr AMAD Mourad	Professeur	Univ. de Bouira	Examinateur
Mr AKILAL Karim	MCB	Univ. de Bejaia	Invité

Année Universitaire : 2024/2025

Dedication

I would like to dedicate this thesis

To my loving father and mother, whose wisdom, strength, and unwavering support continue.

To my dear brother, my cherished sisters, and their wonderful children, for their unwavering support and encouragement.

To my incredible family, whose unconditional love, boundless support, and constant encouragement have been my greatest source of strength.

To my beloved grandmother and my entire extended family, for their love and prayers.

To my dear friends, whose invaluable presence, encouragement, and support have made this journey more meaningful.

Acknowledgments

Praise be to Allah, who enlightened us the path of science and knowledge, bestowed upon me resilience and bravery, instilled in me a passion for science and the virtue of patience, and granted me the strength to fulfill this duty and complete this work.

I would like to express my sincere gratitude to my supervisor Pr. Mawloud Omar. Thank you for giving me an opportunity to work under your supervision and be part of your research group. Thank you for your patient and wise guidance and inspiration throughout my thesis period. I highly appreciate your motivation, supportive attitude, as well as your assistance in disseminating research results, along with your unwavering encouragement and support in my research work, which helped me gain confidence and the courage to persevere and overcome difficulties. Thank you for leading by example and inspiring me with your dedication to research and knowledge. Your guidance helped me evolve and improve in the right direction. Finally, I am so grateful for your valuable insights, suggestions, and the generous time you dedicated to my thesis and paper writing.

I'm very grateful too to DR. Karim Akilal, I sincerely thank him for all his constructive feedback on my research work. Thank you for your patience, generosity, and constant support and motivation during a significant stage of my PHD process. You have helped me develop both technical skills and also as a human being during this crucial phase. For that I am so grateful to you.

A big thank you to my research collaborators, Dr. Siham Bouchelaghem and Dr. Sofiane Aissani, for their expertise and creating a great working environment.

I would like to express my sincere gratitude to the members of the jury Dr. BOUKREDERA Djamila, Dr. FARAH Zoubeyr, Pr. AMAD Mourad for their time, for their time and effort devoted to reviewing my thesis work, as well as for their insightful feedback and constructive remarks, which greatly contributed to its improvement

To all my colleagues, thank you! You have been like a family away from my family. A

special word of thanks also goes to my friends Dalila, Amina, Wahiba, and Amel for their hospitality towards me encouragement, support, and for being a great source of motivation for me.

To my colleagues at the University of Boumerdes, thank you! A special word of thanks also goes to my colleagues in the Economics Department. Thank you to everyone who has helped me, whether near or far.

To all my cool and awesome friends, thank you! a special thanks to Amel and Kahina who has been patient with me, enduring my challenges and moods.

Finally, A big thanks to my family who never doubted or question me, with your support I feel like I can do anything!. A special thanks to my incredible family, who have great credit for this work. I am especially grateful and indebted to my father, mother, brother, and sisters for their unwavering support during difficult times. Their belief in me, along with their prayers and emotional encouragement, has been a constant source of strength. Without their unwavering faith in me and their constant support in pursuing my dreams, I would never have been able to complete my thesis. Words are not enough for expressing my gratitude to them.

Contents

List of Figures viii

List of Tables ix

1 Introduction 1

2 The Fifth Generation Networks 5

2.1 Introduction . . . 5

2.2 The Fifth Generation Networks . . . 6

2.2.1 Mobile communication system generations . . . 6

2.2.2 Main 5G evolution stages . . . 7

2.2.3 Network softwarisation in 5G networks . . . 9

2.2.3.1 SDN (Software Defined Networking) . . . 9

2.2.3.2 NFV (Network Function Virtualization) . . . 10

2.2.3.3 SDN and NFV binding role on 5G networks . . . 11

2.2.4 Network slicing . . . 11

2.3 Internet of things . . . 14

2.3.1 Brief history . . . 15

2.3.2 Definitions of IoT . . . 16

2.3.3 IoT under the 5G cellular networks . . . 18

2.4 Conclusion . . . 20

3 Predictive Maintenance for QoS in 5G Communication: State of the Art 21

3.1 Introduction . . . 21

3.2 Mobility modeling for network optimization . . . 22

3.3 Main mobility prediction methods . . . 23

3.4 Mobility prediction in 5G . . . 24
3.4.1 History-based prediction methods . . . 25
3.4.2 Measurement-driven prediction methods . . . 26
3.5 Mobility prediction helps predictive maintenance . . . 27
3.5.1 Traffic and Bandwidth Management . . . 27
3.5.2 Cross-layer and network slicing optimization . . . 29
3.5.3 Overall Discussion . . . 30
3.6 Conclusion . . . 31

4 Next-Cell Prediction with LSTM based on Vehicle Mobility for 5G mc-IoT Slices 34

4.1 Introduction . . . 35
4.2 Context and motivation . . . 35
4.3 Mission critical IoT communication . . . 40
4.3.1 Applications, requirements and assumptions . . . 40
4.3.2 Mobility support . . . 41
4.4 Network slicing architecture model . . . 42
4.5 AI-Mobility Prediction for mcIoT Slice Enhancement . . . 43
4.6 Next-Cell awareness in 5G network slicing system . . . 45
4.6.1 Next-Cell Prediction for Slice Maintenance under Mobility . . . 48
4.7 The proposed solution . . . 49
4.7.1 Problem definition . . . 49
4.7.2 LSTM Preliminaries . . . 50
4.7.3 LSTM Framework for next cell prediction . . . 52
4.7.3.1 The preprocessing phase . . . 54
4.7.3.2 The classifier framework description . . . 54
4.7.3.3 The training and prediction phases . . . 56
4.8 Conclusion . . . 59

5 Performances Evaluation 60

5.1 Introduction . . . 60
5.2 Dataset generation . . . 61

5.3 Experiment settings 64
5.3.1 Datasets splitting for training and evaluation 64
5.3.2 Evaluation metrics 64
5.3.3 Models hyper-parameters tuning 66
5.3.3.1 Hyperparameter selection for the LSTM model 67
5.3.3.2 Hyperparameter selection of traditional ML models 67
5.3.4 Results and analysis for each dataset 67
5.3.4.1 Results of the LSTM classifier 68
5.3.4.2 Results of traditional ML classifiers 70
5.3.5 Results and analysis using multi-classification metrics 76
5.3.5.1 Model evaluation using confusion matrix 76
5.3.5.2 Model evaluation using F1_Score, precision , and recall metrics 78
5.3.5.3 Discussion 78
5.4 Practical integration of the proposed approach 80
5.4.1 Scalability concerns 80
5.4.2 Computational requirements 81
5.4.3 Potential challenges 81
5.5 Conclusion 82

6 Conclusions and Future Works **83**

List of Figures

2.1 From 1G to 5G :Evolution of Cellular Networks 6
2.2 The evolution stages of 5G networks under IMT-2020 standards 7
2.3 SDN Architecture [1, 2] 9
2.4 NFV Architecture [3, 4] 9
2.5 The Three 5G slice groups 13
2.6 Evolution of IoT over the past decades 15

3.1 Mobility Prediction in 5G 25
3.2 Mobility Prediction for Resource Management in 5G 28

4.1 The different mc-IoT application scenarios and requirements 41
4.2 Network slicing architecture. 43
4.3 PdM strategies for enhancing 5G mcIoT slice QoS. 48
4.4 Our proposed next-cell classifier. 53
4.5 Dataset split into training and testing sets. 57
4.6 Training phase to find f^* for a given iteration. 57
4.7 Testing phase for the next-cell prediction. 58

5.1 Area of the mobility simulation. 61
5.2 Collected mobility information profiles. 62
5.3 Overall constructed datasets. 63
5.4 LSTM model loss and accuracy on test data versus history length. 68
5.5 LSTM model loss on training and testing data for each $\texttt{dataset}_k$. 69
5.6 LSTM model accuracy on training and testing data for each $\texttt{dataset}_k$. . . . 70
5.7 KNN model loss and accuracy on test data versus history length. 71
5.8 KNN model loss on training and testing data for each $\texttt{dataset}_k$. 71

5.9 KNN model accuracy on training and testing data for each $\texttt{dataset}_k$. 72
5.10 RF model loss and accuracy on test data versus history length. 73
5.11 RF model loss on training and testing data for each $\texttt{dataset}_k$. 73
5.12 RF model accuracy on training and testing data for each $\texttt{dataset}_k$. 74
5.13 SVM model loss and accuracy on test data versus history length. 75
5.14 SVM model loss on training and testing data for each $\texttt{dataset}_k$. 75
5.15 SVM model accuracy on training and testing data for each $\texttt{dataset}_k$. 76
5.16 LSTM Confusion Matrix. 77
5.17 KNN Confusion Matrix. 77
5.18 RF Confusion Matrix. 78
5.19 SVM Confusion Matrix. 78
5.20 Comparison of the LSTM classifier with other ML methods. 79
5.21 Prediction Execution Time per one_sequence. 82

List of Tables

3.1 Mobility prediction helps predictive maintenance 32

4.1 Notations 49

5.1 Hyper-parameter settings for each model 66

5.2 LSTM model results with different sliding windows k 68

5.3 KNN model results with different sliding windows k 70

5.4 RF model results with different sliding windows k 72

5.5 SVM model results with different sliding windows k 74

5.6 Performance of LSTM classifier and the traditional ML classifiers 78

5.7 Accuracy per profile 80

The author publications

Journal papers

Belhadj, A., Akilal, K., Bouchelaghem, S., Omar, M., Aissani, S. (2024). Next-cell prediction with LSTM based on vehicle mobility for 5G mc-IoT slices. *Telecommunication Systems, 87(3), 809-833.*

Conference papers

Belhadj, A., Omar, M, Aissani, S. (2025). Predictive Maintenance for QoS in 5G Communication: A State of the Art Review. *In proceedings of the Tenth International Conference on Information and Network Technologies (ICINT). Melbourne, Australia.*

Chapter 1

Introduction

The Internet of Things (IoT) is regarded as an extension of the traditional Internet by integrating heterogeneous networks composed of constrained and autonomous devices into the global computer network. But, facing the distinct particularities of IoT systems, the research community, as well as the industry, have been actively working on the adoption of IoT systems with the Internet. As that contributed to the appearance of new communication mechanisms to enable and facilitate connectivity among an unprecedented number of smart objects deployed in the world. However, with the significant increase in the number of users generally and objects in particular, and with the volume of data generated by these objects in increasing day by day, have led to increased network congestion [5] and more difficulties to the guarantees for Quality of Service (QoS), as objects may have widely varying QoS requirements, operate on diverse protocols, and support a vast number of applications [6]. Thus, the challenge becomes more and more demanding in order to support QoS and the large scale factor of the delivery of data services in IoT communication system. Due to these aspects and the strong heterogeneity of objects, the design of innovative solutions provided by the fifth generation (5G) cellular network in IoT systems represents the key issue making the main objective of the research work of this thesis.

5G has recently attracted a lot of attention as a potential solution to the adopted of IoT ecosystem. 5G mobile network is designed to provide new and enhanced services that make life easier in several areas introduced in the IoT, like health care, agriculture, transportation systems, smart cities, and manufacturing [7]. As these areas are critical, they are more sensitive to the reliability of communication, which directly impacts the requirement for high-performance maintenance. The 5G has the capability to support multiple combinations

of requirements such as reliability, latency, throughput, positioning, and availability at once [8]. That enable supporting the heterogeneous QoS requirements of users on-demand and in real-time [9]. As same, 5G contributes to overcome the stringent challenges posed by IoT communication system. That provides key enabling technologies for ubiquitous deployment of the IoT technology, and enabled large-scale deployment of IoT applications. To meet the highly heterogeneous network and fulfilling the various QoS demands generated due to the incorporation of a wide range of IoT applications into 5G network, the wireless communications has been split into three slice groups, namely eMBB for enhanced Mobile Broadband, and two other class groups specified for Machine Type Communications MTC use cases, namely mcMTC communication (mc-IoT) for mission-critical communications and mMTC communication (m-IoT) for massive machine-type communications. These three slice groups cover different applications and requirements. In this thesis we focus specifically on the mission critical communication of IoT application scenarios which is intended for time sensitive communications with focus on predictive maintenance readiness. In the mc-MTC use-case, the main challenges are to ensure stable communication with mobility of users while keeping ultra high reliability and ultra low latency communications (URLLC). Whilst, these devices are not focused about features such as high data rates, or massive number of devices.

This thesis examines the role of mobility within the 5G infrastructure, with a particular focus on how mobility prediction can enhance predictive maintenance strategies. By anticipating user movement patterns, mobility prediction enables proactive interventions to mitigate potential service disruptions and performance issues before they arise. Integrating mobility prediction with predictive maintenance enhances resource allocation efficiency while ensuring seamless service delivery across various applications. Our study explores the intersection of mobility management and predictive maintenance, emphasizing the transformative impact of leveraging mobility data to sustain system reliability and optimize real-time 5G operations. While this thesis does not delve deeply into post-deployment maintenance, its primary aim is to facilitate the integration of predictive maintenance methodologies within the 5G ecosystem by the proposition of a new framework of user mobility prediction.

Mobility prediction techniques are potential solutions that can be used to empower network slicing systems towards effective dynamic network slicing and, thus, be able to respond to network variability in near real-time. Therefore, in this thesis we focus on providing move-

ment predictability in mcMTC networks to avoid service degradation of critical services due to users' mobility, which is a promising solution to deal with the dynamic nature of the network. Furthermore, it will enhance the 5G network performance to be able to manage these network slices in advance. Hence, in the context of mcMTC use cases with high mobility support, how to develop and choose the best mobility prediction model, and how can it be used as part of a 5G network slicing architecture to handle critical MTC communications, are the main issues that will be explored in this dissertation.

The main objective of the contribution consists of the development of a next cell prediction framework based on LSTM, that enhance the 5G network slicing systems by intelligent and proactive decisions making to maintain the seamless connectivity of mc-IoT slices in a dynamic environment, while ensuring the service quality. Therefore, our research focused on high mobility mc-IoT applications that need ultra reliable and low latency communications. In which, the proposed framework exploits the collected historical mobility traces of crossing machine-type users of these applications, and the LSTM Deep Neural Network (DNN) for the prediction of the future serving cell for a given user. In which, next-cell awareness in 5g network slicing systems can be harnessed to perform suitable actions for mc-IoT slices in advance to avoid service degradation and to ensure the URLLC QoS over these slices. We enumerate the major contributions of this thesis:

1. We perform a comprehensive state-of-the-art review, highlighting mobility-driven predictive maintenance as a key enabler of reliable, real-time 5G operations.

2. We propose an LSTM-based next cell prediction framework, which is a classification process using a many-to-one scenario and has been treated as spatial time series forecasting problem. The input data are multi-variate features including the user-profile identity and locations that are already visited by the user in the past.

3. We perform a real-trace-driven case study for vehicular networks to demonstrate the performance of the proposed classifier. We generated realistic vehicles' trace data using SUMO [10]. We explored the generated mobility data to extract a trajectory representation datasets as a time series of locations with different sliding windows.

4. We conduct experiments on the different generated users trajectories datasets to provide a deep analysis of the proposed approach on short-term mobility prediction, with

high precision results.

For the ease of the reader, the content of the following chapters is summarized as follows. **Chapter 2** examines the current research on both 5G and IoT domains, it is indeed to explore IoT from the viewpoint of cellular telecommunication systems. **Chapter 3** deals with the background and related work that tackle the MTC communications within the 5G systems using different classification criteria. Then, it introduces a state-of-the-art on prediction techniques and their applications in 5G to support ultra reliable and low-latency class services and predictive maintenance. **Chapter 4** explains the range of mobility mcMTC applications supported by the 5G network, the adopted network slicing architecture, and the detailed description of the proposed approach. **Chapter 5** presents the performances evaluation. The analysis is performed on the efficiency of the proposed next-cell framework within the scope of mc-IoT vehicle communication. **Chapter 6** concludes this thesis by summarizing the major findings. It provides an outlook on potential future research directions.

Chapter 2

The Fifth Generation Networks

Contents

2.1 Introduction 5

2.2 The Fifth Generation Networks 6

2.2.1 Mobile communication system generations 6

2.2.2 Main 5G evolution stages 7

2.2.3 Network softwarisation in 5G networks 9

2.2.4 Network slicing 11

2.3 Internet of things 14

2.3.1 Brief history 15

2.3.2 Definitions of IoT 16

2.3.3 IoT under the 5G cellular networks 18

2.4 Conclusion 20

2.1 Introduction

Cellular networks and Internet of Things (IoT) began as separate technologies, independent from the Internet—one focused on mobile voice calls, the other on connecting objects. Over time, they became an integral part of the Internet. Cellular networks started becoming a crucial part of the global Internet infrastructure with the introduction of the third generation in 1998. Meanwhile, the origins of IoT can be traced back to 1999, when RFID technology

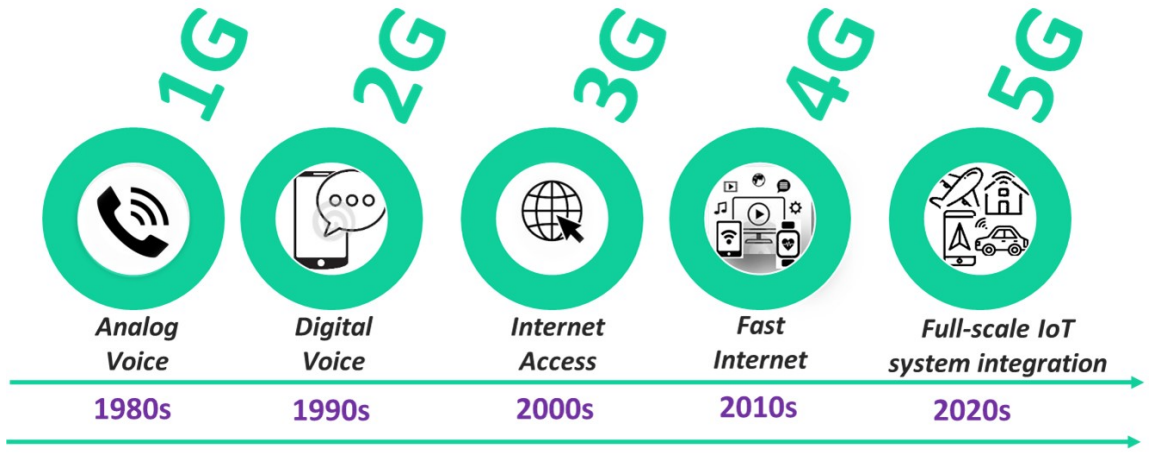

Figure 2.1: From 1G to 5G :Evolution of Cellular Networks

was integrated into networking by linking objects to the Internet using tags. A decade later, the integration of IoT systems into 5G mobile networks became inevitable under the specifications of the IMT-2020 project. This is because 5G has proven to be superior in addressing IoT needs better than traditional IP-based solutions, thanks to several key advantages specifically designed to meet IoT requirements. These advantages have enabled large-scale IoT deployment, overcoming critical challenges such as device diversity, mobility, security, and scalability. In this chapter, we present current research in both the 5G and IoT domains, exploring IoT from the perspective of cellular telecommunication systems.

2.2 The Fifth Generation Networks

5G network is the first generation of mobile networks specifically designed to meet the diverse requirements of various vertical industries. In this section, we talk about the most important concepts and components of 5G networks.

2.2.1 Mobile communication system generations

Mobile network generations have undergone five distinct phases of evolution as illustrated in Figure 2.1. The first generation (1G) of mobile communications system was based on analog transmission for voice services, and led to the creation of the first cell phone [11]. The second generation (2G) outperformed the first generation systems in different functionalities. 2G systems provided service text messaging, international roaming, and automatic location services, while other services are appeared the 2.5 generation like Multimedia Messaging

Service, instant messaging and others [12]. The third generation (3G) was envisioned as a system that satisfies International Mobile Telecommunications-2000 specifications by enabling mobile Internet connectivity [11]. Next, the fourth generation (4G) has delivered the speed of traffic we enjoy today, as it has proven its effectiveness and is now part of everyday life [5, 12]. 4G networks has succeeded by increasing its network capacity to render users and devices be able to connect to them. It is advantages compared to previous generation by including higher bandwidth and data speeds, lower latency, higher network capacity, easier network integration through the IP network, and enhanced security [5]. Further, 4G started by the precursor to 4G technology due to the development of new technologies and enhancements such as added new frequency bands, enhanced support for heterogeneous networks, as well new technological solutions have been designed specifically to support massive IoT [12]. Beyond the 4G, 5G represents the next major phase in mobile telecommunication standards. 5G is designed to accommodate the increasing number of users, high number of demands, the emergence of the IoT, and new applications that require very critical low latency and very high capacity, and the heterogeneous QoS.

2.2.2 Main 5G evolution stages

This section presents the evolution stages of 5G networks under IMT-2020 standards, as displayed in Figure 2.2.

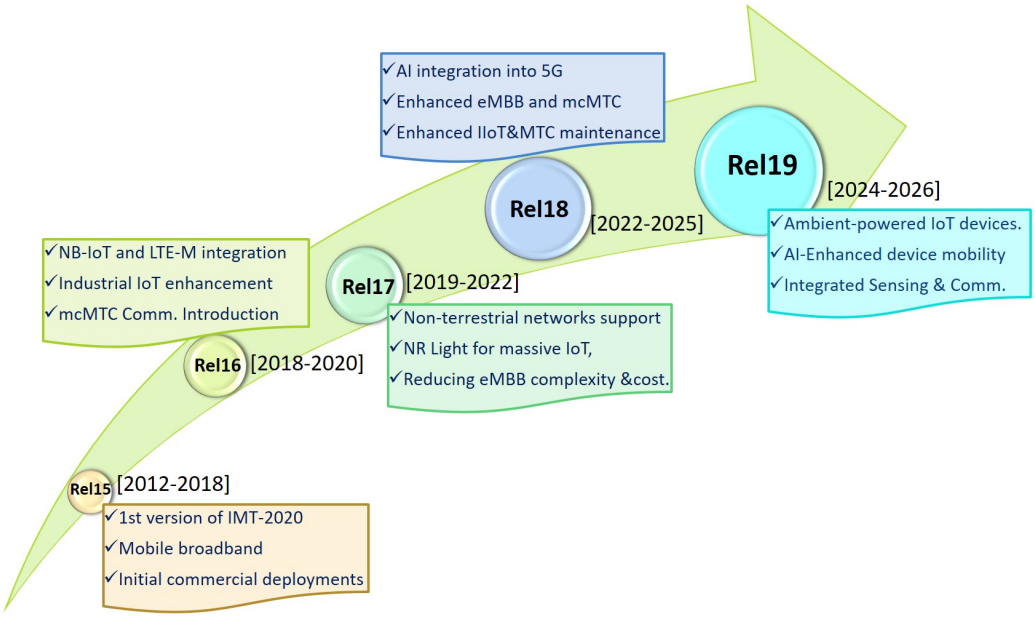

Figure 2.2: The evolution stages of 5G networks under IMT-2020 standards

Below, we summarize the key milestones of the 5G stages [13, 14, 15, 16]:

1. **Release 15 (Rel-15)** (2012-2018):
    - Initial commercial deployments of 5G began in 2019.
    - Focused on mobile broadband communication, battery-efficient security for low-throughput devices.
2. **Release 16 (Rel-16)** (2018-2020):
    - Integrated NB-IoT (NarrowBand Internet of Things) and LTE-M (Long-Term Evolution for Machines) into 5G for dedicated MTC (Machine Type Communication) services.
    - Enhanced support for the Industrial Internet of Things (IIoT).
    - Introduced mission-critical communication for use cases requiring high reliability.
3. **Release 17 (Rel-17)** (2019-2022):
    - Expanded 5G reach to support non-terrestrial communication (e.g., satellites and drones).
    - Developed "NR (New Radio) Light" for massive IoT, balancing higher data rates and lower latency compared to LTE-M while reducing complexity and cost relative to eMBB (enhanced Mobile Broadband).
4. **Release 18 (Rel-18)** (2022-2025):
    - Integrate of artificial intelligence to offer smart and data-based solutions to the 5G network, in order to enhance network functionalities such as network energy saving, load balancing, and mobility optimization.
    - Address additional requirements from mobile operators and verticals such as eMBB and mcMTC (Mission Critical MTC) security enhancements.
    - Enhance the IIoT and the maintenance of MTC.
5. **Release 19 (Rel-19)** (2024-2026):
    - Focused on integrating ambient power-enabled IoT devices.

- Enhanced 5G device mobility using AI-driven solutions.
- Conducted studies on Integrated Sensing and Communication.

2.2.3 Network softwarisation in 5G networks

Network softwarisation and virtualization technologies aim to provide cellular network with greater flexibility, programmability, and adaptability. Among which, SDN (Software Defined Networking) and NFV (Network Function Virtualization) tools used in 5G network.

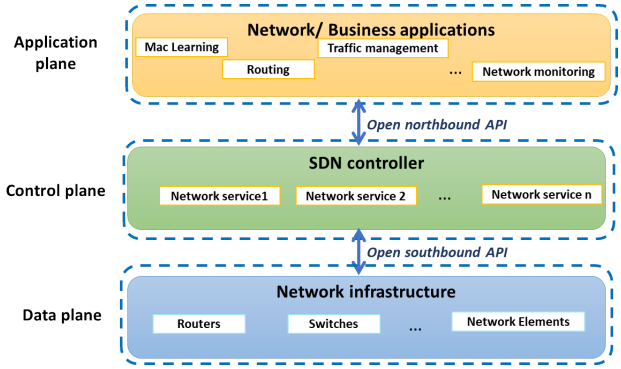

Figure 2.3: SDN Architecture [1, 2]

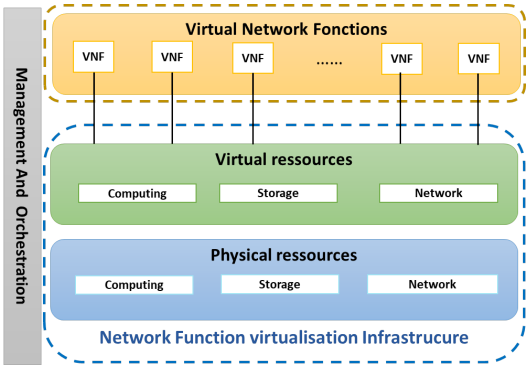

Figure 2.4: NFV Architecture [3, 4]

2.2.3.1 SDN (Software Defined Networking)

The Open Network Foundation (ONF) defines SDN as "the physical separation of the network control plane from the forwarding plane, where a control plane manages multiple devices" [17]. SDN [18] involves decoupling the control plane from the data plane to enable programmable and dynamic network configuration through open APIs. A centralized controller within the control plane is responsible for configuring the forwarding devices, which operate based on programmable packets. Figure 2.3 depicts the SDN architecture with three layers, as outlined in the ONF reference model [19]. It illustrates the separation of network control and forwarding functions, enabling centralized and direct control of the network via open APIs. The central component of this architecture is the SDN controller, a server located at the core of the system. The SDN controller acts as the intelligent entity managing the resources of the data plane to deliver services. Its primary role is the real-time, multi-dimensional optimization of resources and service demands in a dynamic environment, adjusting to changing criteria over time [19].

The architecture, based on open APIs and SDN functionalities, can be divided into

top-down and bottom-up components. The top-down component ensures service delivery, where operations across the SDN NorthBound Interface (NBI) involve service invocation and management between the application layer and the control layer [19]. Through the NBI, SDN applications specify their traffic management requirements in the underlying networks [1]. This interface is also referred to as the intent interface [19]. The SDN controller in the control plane validates requests against policy and resource availability. It then either satisfies these requests by translating application requirements into forwarding rules for the underlying network switches [1] or responds with appropriate exceptions [19].

In contrast, the bottom-up component involves the SDN controller managing the resources of the data plane through the SouthBound Interface (SBI). The data plane consists of network elements, such as switches and routers, that process packets according to rules provided by the SDN controller. These elements also collect network status information, such as topology and traffic statistics [1]. Depending on the rules installed by the controller, the SBI interface enables the controller to configure devices to function as routers, switches, firewalls, or perform other roles, such as load balancing, reporting network status, and managing packet forwarding rules [18].

2.2.3.2 NFV (Network Function Virtualization)

Telecommunication networks leverage the so called Network Functions (NFs) to operate on their traffic. For instance, network functions that collect statistics, modify packet headers, load balancers, or drop packets matching patterns of malicious traffic [20]. NFV is a network architecture framework where the physical NFs are implemented in software [2]. Thus, NFV technology decouples the hardware NFs from dedicated hardware, making it possible to implement the network functions to generic software running on standard platforms like switches, servers, and storage [21]. The main tasks of the NFV are [22]:

1. Separation of software from hardware: it enables the software to be separated from the hardware

2. Flexible deployment of network functions: NFV is automatically able to deploy network function software on a set of hardware resources that may run different functions in different data centers at different times.

3. Dynamic service provisioning: NFV is automatically able to deploy network function software on a set of hardware resources that may run different functions in different data centers at different times.

NFV technology [5] was originally developed by leading service providers to facilitate the shift from hardware-centric to software-oriented infrastructure, that accelerates the deployment of new network services, fosters revenue growth, and simultaneously reduces operational costs, ensuring a more agile and cost-effective network environment.

2.2.3.3 SDN and NFV binding role on 5G networks

As explained in [2], NFV and SDN are distinct technologies, but they are highly complementary. Combining them in a unified networking solution can deliver greater value, particularly in 5G networking systems. On one hand, the centralized control and management applications of SDN provide a global perspective of network slices. On the other hand, NFV technology enhances reliability, scalability, and elasticity—capabilities that SDN alone cannot achieve. Moreover, NFV and SDN in 5G networks aim to automate the operation of network slices, regardless of their type, while virtualizing the resources and functions required by each slice. Currently, the European Telecommunication Standards Institute (ETSI) has developed the NFV Management and Orchestration (NFV-MANO) framework to efficiently manage and orchestrate VNFs [23]. The ETSI-NFV framework has been widely adopted in the 5G telecommunications industry and is designed to meet key NFV specifications and requirements, including security, scalability, and reliability.

2.2.4 Network slicing

The architecture of 5G network is based on concept of network slicing, allowing each user to be served uniquely according to their specific requirements. That is being a relevant solution for IoT applications characterized by the existence of services with divers QoS requirements.

Due to the integration of a wide range of IoT applications into a 5G network, new use cases were appeared and characterized by different requirements, like latency and bandwidth. That represented a key challenge for network operators to simultaneously fulfill all requirements [20]. For this reason, the 3rd Generation Partnership Project (3GPP) has proposed the technique of network slicing; in which 5G network be able simultaneously to serve users

having different requirements and shared the same physical infrastructure. Moreover, network slicing is regarded as an extension of network sharing, by allocating the required and suitable amount of demanded physical resources for all services of different types, all while running in the same time and under the same physical infrastructure [24]. Furthermore, authors in [20] have defined the network slicing technique as the process of multiplexing logically independent networks on a substrate where each network is dedicated for a single use case. That means the ability of the creation and the management of multiple logical networks known as "network slices" running on top of a shared physical infrastructure, where the objective is to ensure seamless end-to-end service connectivity to different users according to their own QoS requirements. In addition, the network slicing process under the specification of 5G networks is characterized by a set of criterion need to be taking into account such as automation and adaptation, elasticity and scalability, isolation, programmability, customization, hierarchical abstraction, and optimization.

5G network slicing faces mobility management challenges due to the increasing number of smart devices and the diverse requirements of vertical industries. Each 5G network slice has unique characteristics and requirements concerning mobility and latency. For instance, the mobility management and handover support needed for automated driving services differ significantly from those required for mobile broadband slices. High-speed trains, for example, can trigger multiple handovers within a short period for railway communications in 5G networks. Fast handover with seamless mobility support is essential for real-time services such as multimedia streaming. However, not all network slices require mobility management support in 5G. For example, network slices designed for industrial control do not need mobility management functions because the devices involved are typically fixed in position. Recent studies have investigated mobility management and handover mechanisms in 5G network slicing [17].

In the following we list each slice type by citing its role, characteristics, and the main application scenarios it serve (see Figure 2.5):

- **eMBB slice** is reserved for mobile communication, addressing human-centric use cases for access to multimedia content, services and data [25]. It focused to enhance users Quality of Experience (QoE) on existing mobile broadband business scenarios [26]. It is characterized by high capacity of bandwidth and high capabilities of multimedia

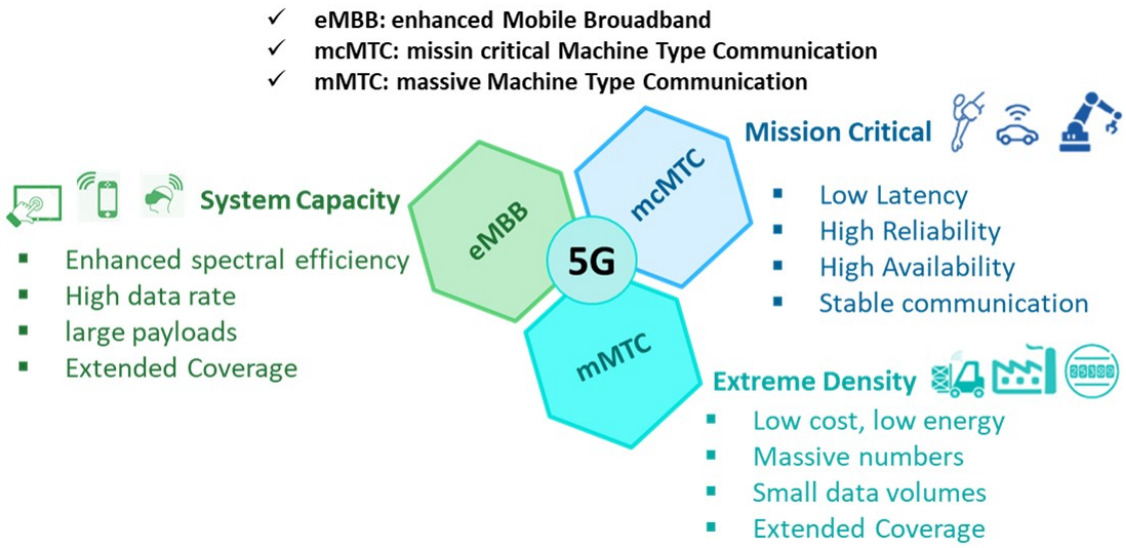

Figure 2.5: The Three 5G slice groups

streaming processing, including handling of three dimensional video, virtual reality, hologram and other related services [27], that needs a high data rate can be exceed 10 Gb/s for eMBB services [28]. Hence, the slice management and orchestration is requested to assign efficient virtual I/O ability and processing ability of data plane [27]. At the same time, the NFV-based mobile edge computing capability, local caching servers deployed at the edge cloud can offer traffic offloading and latency reduction [29]. The eMBB slice covers many application scenarios such as: smart home, smart building, Virtual reality and augmented reality(VR/AR), and smart devices.

- **mMTC slice** is machine type slice designed for a very large number of connected devices typically transmitting a relatively low volume of non-delay-sensitive information [25]. It's main characteristics are low cost, low energy consumption, and short payload with massive number of connections among IoT devices [26], and consume massive radio and signaling resources [27]. It is intended to support IoT in a wide area radio coverage among humans and objects [26], that entails providing energy and spectral efficient connectivity to a large number of IoT devices [30]. Moreover, the accent here of mMTC is on scalable connectivity for an increasing number of devices, wide area coverage and deep indoor penetration [31]. Therefore, the core network (CN) functions can be carried out at the centralized core cloud for cost effectiveness, and mobility management functionalities can be omitted in the slice [29]. This slice supporting divers services in IoT applications that are typically delay-tolerant and have

no mobility. These applications belonging to massive MTC (mMTC) use case, they able to monitor, respond immediately to user input, which are known as monitoring-oriented applications [32]. It is also about handling sensor networks measurement in various areas of smart cities, smart home and industrial, as typical use cases we cite: smart agriculture, smart metering tracking, and logistics.

- **mcMTC slice** is also a machine type slice, and the most critical slice in the 5G systems. It designed for Ultra-Reliable-Low Latency Communications (URLLC) and covers the timely sensitive application scenarios. It needs a fast creation, modification, migration of virtual network functions and reliable transmission from end to end [27]. Thus, the most core network functionalities will be processed at the edge cloud in order to minimize latency [29]. That requiring an edge cloud and Radio Access Network (RAN) resources of a high performance in order to provide the sufficient virtual resources. These specific network slices are expected to offer agile and on-demand scaling ability to handle abrupt data transmission and processing [27]. Moreover, The main goal of mcMTC slice is to enhance the user's QoE by establishing a highly reliable, low latency and stable communication [33]. The security and availability are also critical in such type of slices. In addition, its data characterized to be burst, and with small payloads [34]. However features such as high data rates, or massive number of connections are not required [35]. This slice designed to serve applications extremely sensitive to latency. These applications [32] are often referred to as control-oriented IoT applications because they can manage external environments by connecting to computer systems through smart input/output devices such as RFID tags, sensors, or actuators. For example we cite some applications: industrial and remote control applications, remote surgery, Internet of vehicles, etc.

2.3 Internet of things

Due to the exponential growth of things connected to the Internet, and also the volume of data generated by IoT devices in increasing day by day, the transfer of these enormous data on the cloud brings a big challenge due to limited bandwidth and overloading. In addition, IoT has become included a wide range of applications with diverse requirements in

many different verticals and industrials. This ranges from best-effort connectivity for simple sensors to high data-rate, highly reliable real-time connectivity for Machine Type Devices (MTDs) like vehicles and drones [36]. Such conditions limits the large scale deployment of IoT applications, and may adversely degradate the performance of the time-sensitive applications, potentially leading to increase latency and inefficiencies and, so, the non-satisfying of the QoS. Consequently, in recent years, the advent of 5G mobile networks has demonstrated their superiority in addressing IoT needs more effectively than traditional IP-based solutions. This is attributed to several advanced technologies and unique advantages that 5G offers, specifically tailored to meet the diverse requirements of IoT. These characteristics drive the evolution of IoT systems toward large-scale deployment, while at the same time overcoming the stringent challenges associated with IoT applications, including device and service diversity, mobility, security, interoperability, scalability, and more.

2.3.1 Brief history

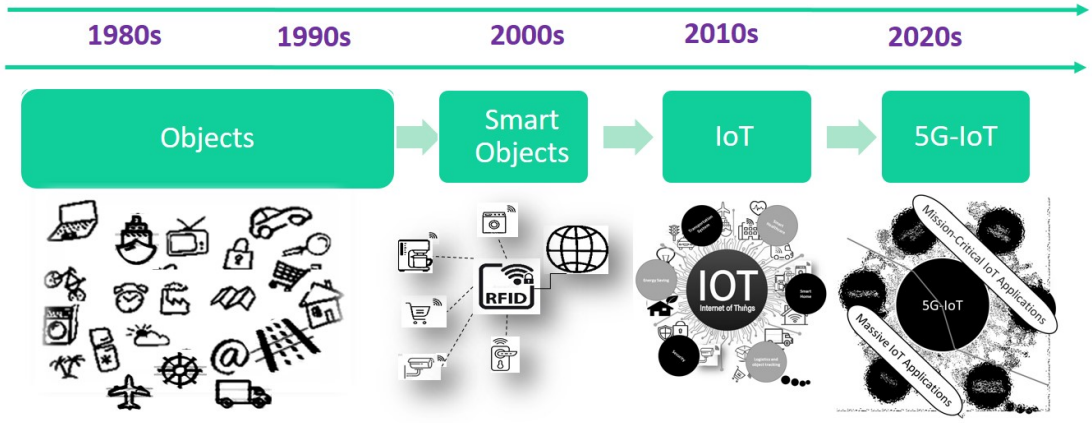

Figure 2.6: Evolution of IoT over the past decades

From the history, the IoT is inspired first by the extensive use of RFID tags in transportation and logistics, then it was progressively generalized to everyday things (smart objects) to achieve a worldwide network of connected devices. The first Internet appliance was a Coke machine at Carnegie Melon University in the early 1980*s* [37]. The term IoT has been coined by Kevin Ashton in a presentation in 1999, with reference to how to use the RFID technology in the Procter and Gamble company for supply chain management [38]. But, according to D. Engels, the term was used in a 1997 publication by the ITU [39]. Then, the concept of IoT first gained traction through the Auto-ID center in 2003 and in related

market analysts publications [37]. In 2005, the ITU published the first technical report that describes the concept of IoT. The ITU formally defines the IoT as a combined part of ubiquitous networks, next-generation networks and ubiquitous computing, according to which: "available anywhere, from anytime, by anything and anyone", where the main objective of IoT is "physical things are connected to the virtual things via the Internet" [40]. While 2010 marked a pivotal point for its expansion and adoption of IoT. Thus the theoretical concept of "objects around us be able to communicate with each other over the Internet" become a reality. Nevertheless, IoT in the context of cellular networks have appeared from the 3G and 4G network generations, but they are not entirely suited for such IoT systems [41]. In contrast, 5G considered as a key enabler for the IoT from 2020s, where the main goal of IoT as mentioned in [42] becoming in simple phrase, "is to plug and play smart objects".

In summary, IoT evolved over decades (see Figure 2.6). Objects did not have Internet access in the 1990s [41]. This era's technologies operated independently of each other. As RFID technology appeared then integrated to networking by linking objects to the Internet via tags in 1999s [43], and as smartphones and televisions started gaining internet access in 2000s, sparking the rise of connected phones. The practical beginning of the IoT being in the early 2010s. Whereas, due to the 5G deployment in 2020s, the 5G technology advancements, such as low latency, massive connectivity, new radio communications and new networking paradigms formed a crucial role in achieving the vision of a global IoT network.

2.3.2 Definitions of IoT

There are no unique, or universal definition. IoT definitions come in different forms and change depending on the context being addressed. Before the advent of 5G mobile networks, the authors in [43] state of the art definitions of IoT offered by standardization organizations. In what follows, we reference few of them:

Definition of ITU: ITU describes the IoT as an ubiquitous network, using the 4A connectivity: *"Anytime, Anyplace, for Anyone, we will now have connectivity for Anything"*. Later in [44], the 6A connectivity is described as: *"IoT is to allow things to be connected Anytime, Anyplace, with Anything and Anyone, ideally using Any path/network and Any service"*. Additionally, ITU described the enabling technologies for the realization of the IoT. Then, ITU-T Study Group for next-generation networks and future networks has for-

mulated the following definition: *"IoT: A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies."*

Definition of Cisco: In 2013, Cisco defined on the IoT under the term "The Internet of Everything," as: "Bringing together people, process, data and things to make networked connections more relevant and valuable than ever before, turning information into actions that create new capabilities, richer experiences and unprecedented economic opportunity for businesses, individuals and countries."

Definition of IETF: The Internet Engineering Task Force (IETF) defines the IoT by addressing the concepts of both "Internet" and "things." According to IETF, "things" in the IoT vision are highly diverse, encompassing computers, sensors, people, actuators, refrigerators, TVs, vehicles, mobile phones, clothes, food, medicines, books, and more. These are categorized into three scopes: people, machines (e.g., sensors and actuators), and information (e.g., clothes, food, medicines, and books). Each "thing" must have a unique identifier to facilitate communication, addressing, and identity verification. Once identified, a "thing" is termed an object. Regarding the "Internet", IETF emphasizes that while the traditional Internet is based on the TCP/IP protocol suite, not all networks using TCP/IP are considered part of the Internet. For IoT, the concept of the Internet expands to include both TCP/IP-based and non-TCP/IP-based networks.

However, the emergence of 5G enhanced largely the data communication in IoT systems, and accommodate its particularities in term of higher level of heterogeneity, limited resources, mobility, and different QoS requirements, etc. Hence, it is indeed to define IoT from the view point of 5G cellular networks, and vice versa.

The IoT expands the legacy cellular networks to accommodate autonomous networks with MTDs. While, the 5G cellular network expands the IoT capabilities to ensure the delivery of its heterogeneous services with high quality. Therefore, IoT in 5G networks represented by MTC networks that formed by a set of smart MTDs that have multiple capabilities like computing, storing, communicating, sensing, and actuating [45]. MTC covers all related technologies necessary to facilitate the connections of MTDs with minimal or without human intervention [46]. As same, MTC characterized by two type of communications to cover the different IoT applications with diverse requirements and verticals. In which, massive-MTC designed for IoT non-delay sensitive application scenarios and provides best-effort

connectivity for simple sensors, while, mission-critical MTC intended for timely-sensitive applications and offers high speed, high data-rate, highly reliable real time connectivity, for vehicles for example [36]. In the other hand, due to the incorporation of IoT applications, 5G network is considered as a complex network with stationary and mobile devices. It integrates constrained and unconstrained devices with both Human Type Devices (HTDs) such as tablets and smartphones, and MTDs such as drones, sensors, vehicles, and actuators [47].

2.3.3 IoT under the 5G cellular networks

The number of connected MTDs surpassing the number of human-centric connections significantly and every year. Hence, the number of data generated by these devices in exponential increase accordingly. This fascinating development is a driving force behind the convergence of the physical and digital worlds that promises to create an unprecedented IoT [48]. IoT becoming the foundation of many industrial and verticals, that is being applied for several sectors reaching all areas of everyday life. It includes diverse range of application scenarios like smart home automation, smart grid and energy management, intelligent transportation systems, smart agriculture, smart metering, public safety surveillance, among numerous other applications.

Accounting for the variety and the rapid growth pace of IoT applications, the main objective of enabling MTC technology of 5G cellular networks is to construct comprehensive connections among diverse MTDs, stationary and mobile devices, as well as other things across extensive coverage areas [48]. MTC enable the wide range of IoT applications from mission critical services to massive spread of devices. Based on MTC of these applications, IoT applications comprises two broad classes :

- **“low-end” massive IoT applications :** is a subgroup of the IoT applications able to monitor, respond immediately to user input, which are also known as monitoring-oriented applications. In high-volume IoT deployments, there are smart sensors that report on a regular basis to the cloud infrastructure [48]. These applications have low cost, low energy consumption, and short payload with massive number of connections among IoT devices [26].

- **“high-end” mission-critical IoT applications :** advanced critical IoT applications,

that are often referred to as control-oriented IoT applications because they can manage external environments by connecting to computer systems through smart input/output devices such as RFID tags, sensors, or actuators [32]. These applications have stringent requirements in terms of communications reliability, availability, and latency [48].

The coexistence of human-centric and machine-type in IoT applications will lead to a large diversity of communication characteristics. Some of these applications [49] can be supported by today's mobile broadband networks and their future evolution. However, some other applications will impose additional and heterogeneous requirements on mobile and wireless communication systems that the 5G will have to support. In recent years, numerous works have been conducted on a number of challenging topics for such 5G IoT, as well as the key criteria of IoT that is 5G enabled [41]:

- High-scalability and perfectly all right systems are required for 5G IoT to allow fine-grained front-haul networking breakdown through NFV.
- Reduced delay is required in 5G IoT services like haptic Web, AR, video gaming, etc.
- With reliability and robustness, 5G IoT necessitates enhanced availability and transition effectiveness for consumers of IoT devices and applications.
- Safety, unlike typical security strategies that safeguard connection and privacy protection, the upcoming IoT payment service, as well as online wallet services, create a greater safety approach to increase information security.
- To handle billions of low-power as well as low-cost IoT systems in 5G IoT, 5G-powered IoT requires reduced energy technologies.
- Connectivity density, a very large number of sensors would be linked together during 5G IoT, requiring 5G to facilitate the effective transmission of messages in a specific time and region.
- Agility, the 5G IoT ought to be capable of handling a large number of device to-device connections while being mobile.
- With increased data speed, future IoT systems like HD streaming content or rather AR would demand greater data rates to obtain satisfactory performances.

2.4 Conclusion

5G is the first generation of mobile networks specifically designed to tackle the diverse requirements of various vertical industries. In first chapter, we have introduced the existing cellular network generations. Then, we highlighted the different development stages of 5G mobile networks by citing the different improvements brought about by the 5G toward the adoption of IoT systems for each stage. Further, to expanding the market to comprise massive IoT applications as sames as critical IoT applications, we have presented the key enabling technologies for enabling the ubiquitous deployment of such applications. In other part of the chapter, we have provided some backgrounds about the IoT and its different related terms. Then, we have explored IoT systems from the standpoint of 5G cellular networks, as well as the key requirements of IoT that is 5G enabled.

Chapter 3

Predictive Maintenance for QoS in 5G Communication: State of the Art

Contents

3.1 Introduction **21**

3.2 Mobility modeling for network optimization **22**

3.3 Main mobility prediction methods **23**

3.4 Mobility prediction in 5G **24**

3.4.1 History-based prediction methods 25

3.4.2 Measurement-driven prediction methods 26

3.5 Mobility prediction helps predictive maintenance **27**

3.5.1 Traffic and Bandwidth Management 27

3.5.2 Cross-layer and network slicing optimization 29

3.5.3 Overall Discussion 30

3.6 Conclusion **31**

3.1 Introduction

As presented in the previous chapter, 5G networks operate in highly dynamic environments, characterized by virtualized services and the need for seamless connectivity to ensure QoS. Mobility plays a pivotal role in this landscape, influencing resource allocation, performance,

and service continuity. This chapter focuses on studying the concept of mobility within the 5G infrastructure and explores how mobility prediction can significantly enhance predictive maintenance strategies. By accurately forecasting user trajectories, mobility prediction enables proactive measures to address potential service disruptions and performance degradation before they occur. The integration of mobility prediction with predictive maintenance not only optimizes resource allocation but also ensures the highest levels of service across diverse applications. Through a state of the art review, this chapter highlights the intersection of mobility management and predictive maintenance, emphasizing the transformative potential of leveraging mobility data to maintain system health and sustain robust performance in real-time 5G operations.

3.2 Mobility modeling for network optimization

Modeling node mobility has become a linchpin in modern networks, as the ability to anticipate or reliably predict a node's future location is paramount for proactive resource allocation and the avoidance of performance bottlenecks. This paradigm reflects the core principles of predictive maintenance: foreseeing potential constraints before they cause service degradation and implementing timely interventions to uphold consistent network functionality. Consequently, accurate mobility models empower network operators, service providers, and system architects to optimally provision infrastructure and orchestrate resources, ensuring continuity and efficiency under ever-evolving operational conditions.

Over the years, mobility modeling research has been greatly influenced by advancements in human mobility studies, broadening its scope to encompass the multifaceted nature of modern mobile devices and networks. Notably, two primary methodological strands have emerged: real trace-based models and synthetic models. Real trace-based models derive from empirical datasets that faithfully depict the mobility behaviors of network users. These datasets come from diverse sources, including communication logs (satellite links, RAN, and Wi-Fi), as well as sensor data from onboard devices and roadside systems (e.g., loop detectors, traffic cameras, radar) [50]. In tandem, user-centric data such as mobile phone logs, credit card transactions, and social media footprints can provide richer context to the mobility analysis [51]. Modern software-centric architectures, exemplified by SDN controllers, further support the centralized acquisition of mobility data, facilitating the collection and in-

tegration of large-scale, multi-modal datasets [9]. Consequently, comprehensive sources like Call Detail Records (CDRs), Global Positioning System (GPS) trajectories, and assorted mobile network traffic traces have become more accessible, powering increasingly sophisticated mobility analytics. Synthetic mobility models, on the other hand, employ mathematical and algorithmic constructs to recreate key features of real-world motion without requiring extensive empirical data. By abstracting away the complexity intrinsic to measured datasets, these models grant researchers the versatility to examine a wide range of hypothetical scenarios, experiment with new network topologies or algorithms, and fine-tune system parameters in controlled test environments. Their theoretical elegance and scalability make them particularly attractive for scenarios in which obtaining or processing real data might be logistically impractical or prohibitively expensive.

Regardless of whether they stem from real trace-based observations or synthetic abstractions, mobility models serve a critical function in anticipating node placement and movement. By reliably projecting the near-future whereabouts of mobile nodes, operators can adopt predictive maintenance strategies that not only minimize disruptions but also enhance overall system performance by pre-allocating bandwidth, positioning edge computing resources, and tweaking caching mechanisms to preempt congestion. Ultimately, robust mobility modeling resides at the confluence of performance optimization and predictive diagnostics, equipping researchers and practitioners with essential insights for architecting resilient, adaptable, and future-proof network infrastructures.

3.3 Main mobility prediction methods

Mobility prediction has been extensively studied in the literature, with a wide array of methods. These methods can be categorized into memory-less predictors and more advanced approaches capable of capturing long-term dependencies in mobility data. Memory-less predictors, such as time-series models, the Kalman filter, and Markov models, are effective for scenarios with limited historical data. However, these methods often fail to exploit the long-term dependencies inherent in mobility patterns, limiting their predictive accuracy in complex, real-world scenarios. In contrast, the advent of Machine Learning (ML) techniques has significantly improved mobility prediction by leveraging large-scale historical datasets. By collecting mobility information over extended periods, ML-based predictors can identify

intricate patterns in user behavior, thereby enabling more accurate predictions [52]. These methods excel in capturing the temporal and spatial dynamics of mobility, offering enhanced performance over traditional approaches.

A diverse range of user mobility prediction techniques has been proposed, including Markov chain (MC), hidden Markov models (HMM), Artificial Neural Networks (ANN), Bayesian networks, and data mining methods. Each approach exhibits distinct strengths and limitations depending on the application context and data availability. MC models are widely favored for their simplicity and effective performance in predicting mobility based on historical transition probabilities. However, more sophisticated models, such as HMM, ANN, and Bayesian networks, often deliver superior accuracy by modeling complex relationships within the data. Despite their advantages, these methods are constrained by their computational complexity, which can pose challenges in resource-limited environments.

Data mining approaches introduce additional dimensions by incorporating external knowledge such as roadmap information, environmental context, or social interactions. While these methods demonstrate potential for enhanced prediction accuracy, they also reveal limitations due to their reliance on supplementary data sources and the increased complexity of integrating diverse information streams.

Over recent years, real-world trajectory data has emerged as a critical resource for training and evaluating mobility prediction models. The availability of large-scale datasets derived from GPS and CDR traces (and other sources) has facilitated the development of robust and scalable models. However, the effectiveness of any prediction method is inherently tied to the quality, granularity, and diversity of the used data.

3.4 Mobility prediction in 5G

By empowering networks to anticipate handovers, optimize resource allocation, and prevent performance bottlenecks, mobility prediction is a critical enabler of reliable and efficient 5G systems. This section explores foundational categories and advanced methodologies, focusing on their transformative impact on 5G cellular networks.

Mobility prediction methods in 5G networks fall into two categories: history-based and measurement-driven strategies. Each category offers unique mechanisms to enhance system responsiveness and reliability while addressing specific challenges in dynamic environments.

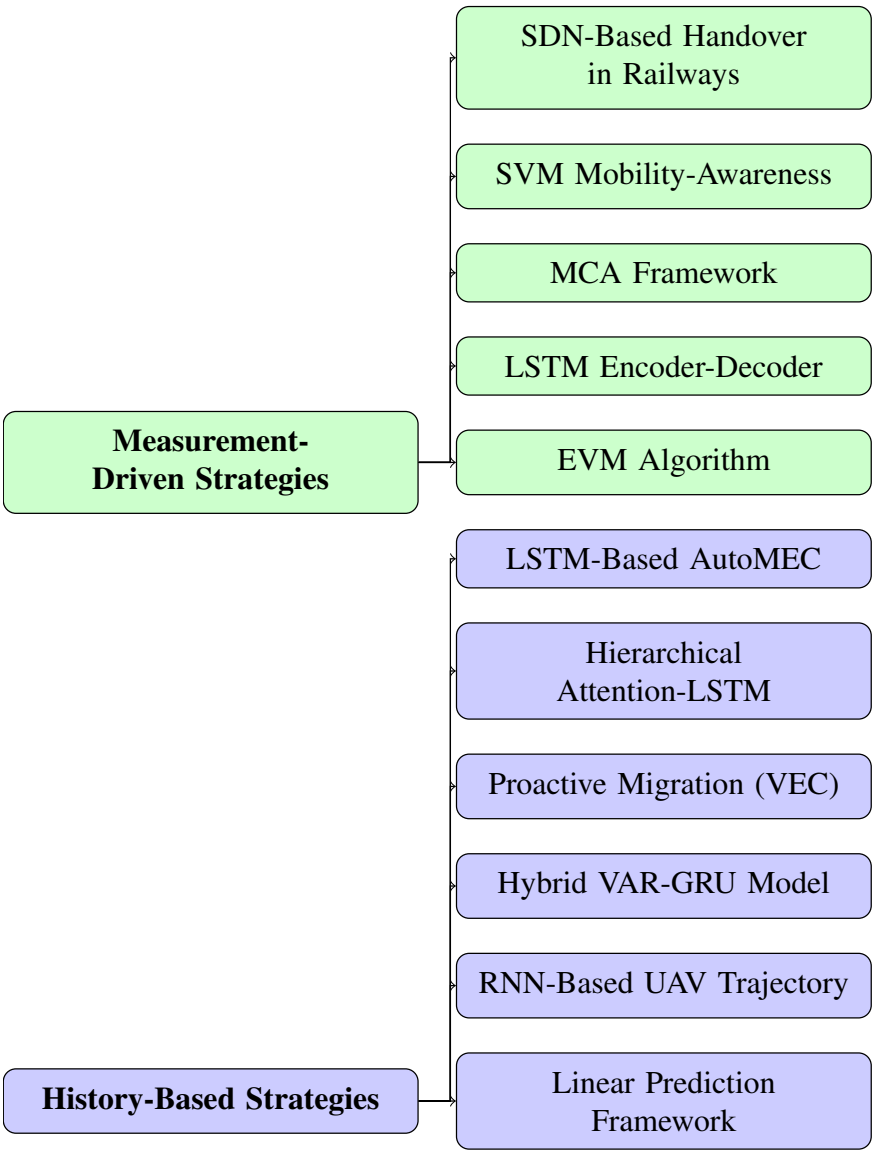

Figure 3.1: Mobility Prediction in 5G

As illustrated in Figure 3.1, we showcase the most significant contributions identified in the literature as part of our analysis.

3.4.1 History-based prediction methods

History-based methods leverage statistical patterns in a user's past mobility records to forecast future movements. By analyzing transition probabilities between network cells, these approaches provide predictive insights into user trajectories. In this context, Hou et al. [53] introduced a linear prediction framework integrating communication co-design, which preemptively transmits device states to minimize latency. However, its dependence on linear systems and higher prediction errors over extended horizons reveal limitations, necessitating advanced models. Xiao et al. [54] proposed an Recurrent Neural Network (RNN)-based

framework for UAV trajectory prediction, combining Direction of Arrival (DOA) estimation with grid-based preprocessing. This method enhances beamforming precision and system reliability but struggles with scalability in highly dynamic scenarios. Similarly, Bahra and Pierre [55] developed a hybrid prediction model integrating Vector AutoRegression (VAR) with Gated Recurrent Units (GRU), addressing challenges like frequent handovers in dense cellular networks. Complementing these efforts, Fattore et al. [56] designed Auto Mobile Edge Computing (MEC), an Long Short-Term Memory (LSTM)-based framework predicting user densities to optimize resource allocation in smart cities and automotive applications. Gilly et al. [57] proposed a proactive migration framework for vehicular edge computing, leveraging mobility predictions to relocate MEC services preemptively. This strategy ensures reduced latency and improved service continuity. Lastly, Li et al. [51] advanced a hierarchical temporal attention-based LSTM encoder-decoder model. By integrating daily and weekly mobility patterns, this approach supports long-term predictions and actionable insights for urban planning and location-based services.

3.4.2 Measurement-driven prediction methods

Measurement-based approaches utilize instantaneous radio parameters (such as signal strength, velocity, and distance) to predict user transitions. These real-time insights complement historical data by capturing current network conditions. In this context, Liu et al. [58] introduced the Edge-Assisted Vehicle Mobility Prediction (EVM) algorithm, integrating Convolutional Neural Networks (CNNs) and RNNs with transfer learning to address challenges like seamless handovers in Vehicle-to-Everything (V2X) communications.

Park et al. [59] proposed an LSTM encoder-decoder model for vehicle trajectory prediction. By employing beam search, their framework forecasts future sequences over occupancy grid maps, enhancing situational awareness in autonomous driving and driver-assistance systems.

In [60], Kuruvatti et al. proposed the Mobility Context Awareness (MCA) framework, which anticipates future Evolved Node B (eNB) associations and proactively migrates MEC services. By incorporating Cooperative Multi-Point (CoMP) transmission and sidelink solutions, this framework ensures uninterrupted connectivity in high-mobility scenarios. Later on, the same authors, in [61], developed a mobility-awareness framework using Support Vec-

tor Machines (SVMs) to predict user trajectories, facilitating efficient resource allocation and enhancing service continuity in vehicular networks.

In [62], Sinha et al. have proposed an SDN-based seamless mobility management framework tailored for Beyond 5G services in high-speed railways. By utilizing Kalman filter-based trajectory estimation and dynamic flow rule installation, it ensures optimized handovers with reduced latency and sustained QoS. The proposed framework enhances the handover efficiency through preemptive connections. However, challenges in scaling to multi-controller SDN environments and reliance on precise trajectory predictions, which may impact reliability in dynamic scenarios.

3.5 Mobility prediction helps predictive maintenance

The following categories illustrate the critical role of mobility prediction in predictive maintenance. By enabling efficient traffic management, dynamic resource allocation, and seamless network slicing, these methods ensure high reliability and low latency in diverse applications. We illustrate in Figure 3.2, the most relevant contributions identified in the literature as part of our analysis.

3.5.1 Traffic and Bandwidth Management

Traffic and bandwidth management play a pivotal role in ensuring that network resources are allocated efficiently to handle fluctuating traffic demands. By leveraging mobility prediction, these methods proactively address congestion and optimize load balancing, ensuring seamless communication for delay-sensitive applications.

Chen et al. [26] have proposed an intelligent traffic-adaptive resource allocation framework for edge-computing-based 5G networks. By employing an attention-enhanced LSTM model for mobile-traffic prediction, the framework forecasts peak traffic flows, enabling proactive resource allocation across edge and remote clouds. Mobility prediction optimizes resource management by dynamically balancing network loads, reducing congestion, and ensuring URLLC. This approach integrates user mobility patterns to enhance resource utilization and improve QoE for delay-sensitive applications in dense networks.

Xiao and Chen [63] have proposed an LSTM-based traffic prediction framework for dy-

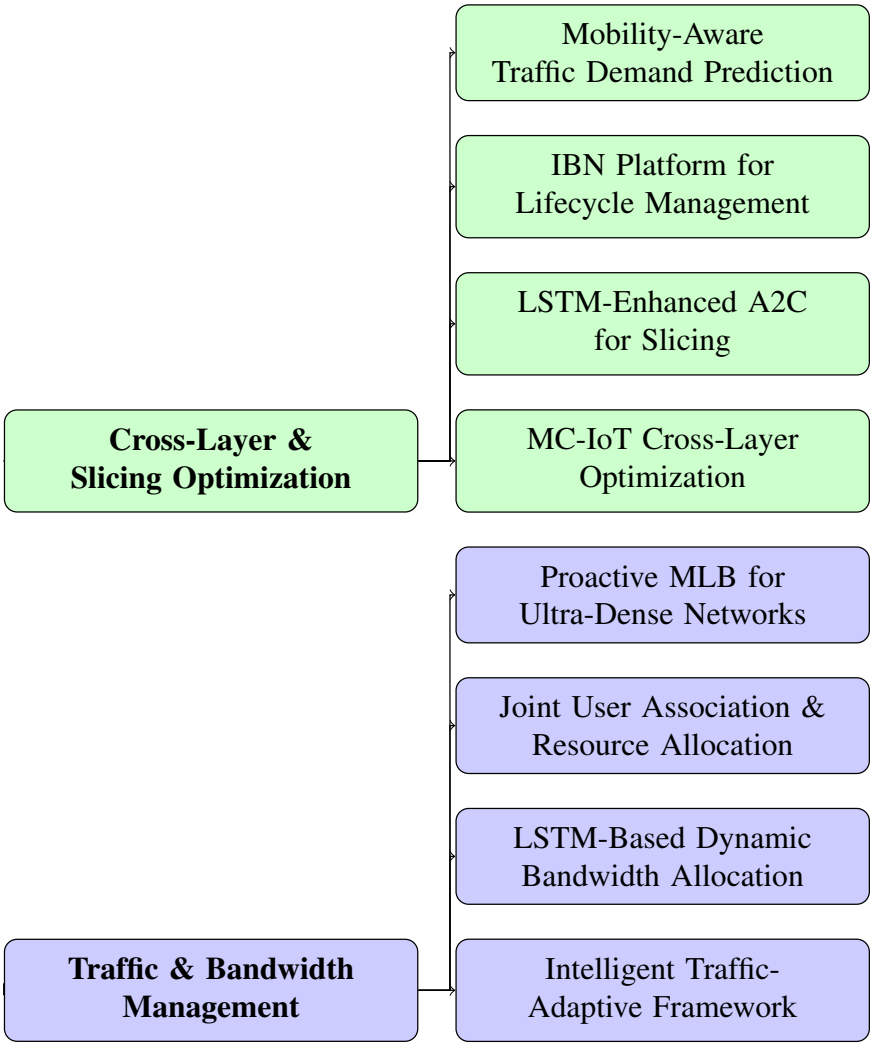

Figure 3.2: Mobility Prediction for Resource Management in 5G

namic bandwidth allocation in 5G transport network slices. The system employs LSTM models to predict traffic demands across different service types. Resource management integrates mobility predictions with a fractional knapsack optimization algorithm, dynamically reallocating bandwidth based on service priorities and traffic patterns. This ensures enhanced resource utilization, reduced packet loss, and improved user experience.

Cheng et al. [64] have proposed a joint user association and resource allocation framework in heterogeneous networks (HetNets) utilizing Virtual Small Cells (VSCs) and mobility prediction. The approach employs MC to predict user mobility. Multi-agent Q-learning and Deep Q-Networks optimize resource allocation by proactively addressing user mobility and interference, enhancing system capacity, spectrum efficiency, and service continuity in complex HetNet scenarios.

Shabbir et al. [65] have proposed a proactive Mobility Load Balancing (MLB) framework for ultra-dense networks, leveraging RNN-LSTM models. The framework predicts user trajectories and access point (AP) loads by analyzing mobility patterns and traffic data, enabling dynamic AP-UE associations. By forecasting AP usage, the system optimizes resource allocation to balance network loads, reduce handover failures, and minimize latency. The proposed framework ensures efficient utilization of network resources while maintaining service quality in dense environments. However, the reliance on accurate trajectory prediction and the computational complexity may limit scalability in large-scale scenarios.

3.5.2 Cross-layer and network slicing optimization

Cross-layer and network slicing optimization focus on harmonizing communication and computational resources across different network layers. By integrating mobility prediction, these methods dynamically adjust resources, ensuring that mission-critical applications maintain their stringent performance requirements.

She et al. [66] have proposed a cross-layer optimization framework for Mission-Critical IoT (MC-IoT) in MEC systems. This framework integrates mobility prediction with resource management by dynamically adjusting user association, packet offloading rates, and bandwidth allocation based on predicted traffic patterns. Mobility insights help balance communication and computation loads, ensuring URLLC. By optimizing edge and network resources, the approach enhances performance for MC-IoT services like factory automation

and remote health monitoring.

Li et al. [67] have proposed an LSTM-enhanced Advantage Actor-Critic (A2C) algorithm for resource management in 5G network slicing. The model incorporates user mobility prediction via LSTM to track dynamic traffic patterns and fluctuating service demands. By leveraging temporal correlations, the algorithm optimizes bandwidth allocation across slices, ensuring high spectrum efficiency and adherence to Service Level Agreements (SLAs). This integration of mobility prediction improves decision-making and supports low-latency, high-reliability communications in complex network environments.

Abbas et al. [23] have proposed an Intent-Based Networking (IBN) platform for managing the lifecycle of network slices in 5G mobile networks. This platform incorporates mobility prediction to optimize resource allocation and ensures QoS compliance by dynamically scaling resources based on predicted usage patterns. The closed-loop mechanism integrates deep learning to automate slice instantiation, monitoring, and updates. This approach enhances resource utilization, minimizes manual configuration, and ensures service reliability in dynamic multi-domain environments.

Tariq et al. [68] have proposed an LSTM-based framework for traffic demand prediction tailored to 5G network slices, focusing on mobility-aware resource management. The system predicts slice-specific traffic demands by analyzing mobility patterns and application usage trends, allowing dynamic resource allocation across the different types of slices. This approach ensures proactive handling of traffic spikes, enhancing service continuity and resource utilization. However, high computational complexity and reliance on precise mobility data limit scalability in dense, heterogeneous networks.

3.5.3 Overall Discussion

This subsection provides a comparative analysis of the discussed mobility prediction methods using six criteria: scalability, mobility prediction precision, resource optimization, Service Quality Assurance (SQA), traffic adaptability, and operational complexity. These criteria reflect core aspects of predictive maintenance, emphasizing proactive resource allocation, reliability enhancement, and operational efficiency. We give in Table 3.1, the overall comparison of the reviewed works. In the following, we give some insights regarding this study.

Scalability is vital for mobility prediction methods to remain effective in dynamic net-

works; for instance, Abbas et al. [23] employed IBN to dynamically adjust resources across multi-domain systems, making it ideal for large-scale predictive maintenance, whereas Shabbir et al. [65] addressed ultra-dense networks with RNN-LSTM models but faced scalability challenges due to computational complexity. Precision in mobility prediction further enables proactive resource allocation: Chen et al. [26] used attention-enhanced LSTM models for accurate peak traffic prediction, improving resource distribution, and Shabbir et al. [65] optimized access point load balancing with trajectory-based RNN-LSTM models. Resource optimization is crucial as well; Li et al. [67] integrated LSTM and reinforcement learning to dynamically allocate resources across slices while meeting strict SLAs, and Cheng et al. [64] utilized virtual small cells (VSCs) to pool resources and address traffic hotspots effectively. Service quality assurance (SQA) is indispensable for reliability and low latency: Xiao and Chen [63] dynamically allocated bandwidth to critical services using fractional knapsack optimization, while She et al. [66] focused on URLLC for mission-critical IoT. Lastly, operational complexity varies across solutions; lightweight approaches such as Xiao and Chen [63] are computationally efficient, whereas more advanced frameworks like Shabbir et al. [65] may face limitations in real-time deployment due to higher demands.

3.6 Conclusion

This chapter has provided a review of how mobility prediction techniques can enhance predictive maintenance strategies in 5G networks. By surveying a variety of modeling approaches, we showcased how accurate forecasting of user trajectories and network states lays the groundwork for proactive service management. Predictive maintenance frameworks become significantly more effective when mobility insights are integrated, enabling preemptive resource allocation, dynamic load balancing, and seamless handover support.

As 5G continues to evolve into a heterogeneous environment with diverse use cases (spanning critical applications such as industrial automation, telemedicine, and vehicular networks) predictive maintenance must evolve accordingly. Advanced data analytics tools, cross-layer orchestration, and real-time network slicing will be paramount in meeting the stringent requirements of low-latency, high-reliability communication. Through this synergy, 5G systems can move closer to self-optimizing infrastructures capable of detecting and mitigating potential disruptions before they degrade network performance or user experience.

Ref.	Scalability	Prediction Method	Resource Optimization	SQA	Traffic Adaptability	Operational Complexity
[26]	Effective in edge-centric networks with manageable complexity	Leverages attention-enhanced LSTM for precise peak traffic prediction	Balances edge resources proactively to minimize latency	Ensures URLLC for delay-sensitive applications	Dynamically adjusts to localized peak traffic fluctuations	Designed for edge-cloud collaboration with moderate computational demands
[63]	Addresses transport network slicing with tailored solutions	Employs LSTM for forecasting traffic demands across service classes	Allocates bandwidth using fractional knapsack optimization	Prioritizes critical services	Implements slice-specific adjustments based on predicted demand patterns	Lightweight greedy algorithm for efficient computation
[64]	Limited scalability in dense networks	Markov-based prediction model for traffic hotspot analysis	Pools virtual small cell resources for dynamic user association	Improves spectral efficiency	Adjusts resource allocation to respond to traffic trends	Relies on multi-agent Q-learning, increasing computational complexity
[65]	Scales in dense networks	RNN-LSTM models for trajectory-based load balancing	Optimizes access point-user associations to prevent congestion	Enhances handover success	Tailors access point load management to dense traffic environments	Computational demands may limit application in large networks
[66]	Targets mission-critical IoT with MEC integration	Integrates mobility to optimize cross-layer resource allocation	Balances communication and computational loads for short-packet processing	Ensures stringent delay for critical IoT services	Adapts to fluctuating IoT traffic patterns with low latency demands	Employs cross-layer algorithms that are moderately complex
[67]	Operates effectively across multiple network slices	Combines LSTM and A2C reinforcement learning for adaptive predictions	Ensures balanced spectrum utilization	Provides robust QoS for diverse applications and traffic types	Proactively manages resources for real-time multi-slice demands	High computational requirements due to deep reinforcement learning
[23]	Scales in multi-domain resource allocation	Utilizes intent-based networking to predict and scale resources dynamically	Automates slice management to optimize resource utilization	Maintains QoS compliance by dynamically scaling the slices	Handles real-time traffic spikes and fluctuating slice requirements	Moderate complexity due to automated closed-loop mechanisms
[68]	Limited in dense networks	LSTM-based framework for traffic demand prediction	Cross-slice dynamic allocation	Service continuity and resource allocation	Proactif slice traffic management	High computational complexity

Table 3.1: Mobility prediction helps predictive maintenance

To complement these efforts and pave the way for enhanced predictive resource allocation, the next chapter will present our proposed mobility model, designed to further optimize network efficiency and reliability under dynamic conditions.

Chapter 4

Next-Cell Prediction with LSTM based on Vehicle Mobility for 5G mc-IoT Slices

Contents

4.1 Introduction **35**

4.2 Context and motivation **35**

4.3 Mission critical IoT communication **40**

4.3.1 Applications, requirements and assumptions 40

4.3.2 Mobility support 41

4.4 Network slicing architecture model **42**

4.5 AI-Mobility Prediction for mcIoT Slice Enhancement **43**

4.6 Next-Cell awareness in 5G network slicing system **45**

4.6.1 Next-Cell Prediction for Slice Maintenance under Mobility 48

4.7 The proposed solution **49**

4.7.1 Problem definition 49

4.7.2 LSTM Preliminaries 50

4.7.3 LSTM Framework for next cell prediction 52

4.8 Conclusion **59**

4.1 Introduction

This chapter is dedicated to our proposition for advancing mobility prediction methodologies. While we do not claim to provide a comprehensive solution for the full establishment of predictive maintenance, we present an innovative approach that achieves high-precision user mobility predictions. This, in turn, facilitates efficient resource orchestration and mitigates performance degradation. Our proposed solution [69] is tailored to a specific use case: vehicular mobility. By focusing on enhancing movement predictability within mcMTC networks, we aim to prevent the degradation of critical services caused by user mobility. This approach addresses the dynamic nature of such networks while ensuring the stringent QoS requirements of URLLC. Moreover, our solution contributes to optimizing 5G network performance by enabling proactive management of network slices, particularly for scenarios requiring robust support for high-mobility mcMTC use cases. Central to this chapter are the following key questions: How can the most effective mobility prediction models be developed and selected? And how can these models be integrated into the 5G network slicing architecture to ensure seamless and reliable machine-type communications? These are the critical issues we explore in this chapter, offering insights into how mobility prediction can empower next-generation networks to meet the demands of dynamic vehicular environments.

4.2 Context and motivation

Cellular systems originated for Human Type Communications (HTC) to serve users with high mobility and high data rate, such as streaming a high-definition video for train passengers [35], which corresponds to mobile broadband communications in 5G communication systems. However, in the context of MTC, mcMTC is the most mobility-sensitive type, where the mobility of connected end-users has a direct influence on the QoS of communications. Accordingly, it is very challenging to satisfy the mcMTC URLLC requirements in a dynamic environment, especially for high mobility applications scenarios cases like V2X communications in 5G enabled vehicular networks.

Network slicing is one powerful tool used to achieve the desired QoS URLLC. It provides the mcMTC slice reserved to handle URLLC. In contrast, for high-mobility application scenarios, a dynamic network slicing can improve the 5G systems to handle the dynamicity

of mcMTC slices when their connected users move. Hence, mobility prediction techniques are potential solutions that can be used to empower network slicing systems towards effective dynamic network slicing and, thus, be able to respond to network variability in near real-time.

The motivation behind this research stems from the greater mobility challenges faced in network slicing for mcIoT compared to eMBB and mIoT slices. These challenges arise from the autonomy of mcIoT applications and their strict latency and reliability requirements. Although network slicing has proven successful in reducing latency and increasing reliability in IoT communications, it still encounters significant limitations in ensuring seamless handovers, maintaining URLLC-level QoS, and guaranteeing stable communications for mcIoT applications. The primary mobility challenges in the mcIoT slice include:

- **High mobility and rapid cell transitions:** Devices in mc-IoT applications often move at high speeds, necessitating frequent transitions between cell base stations. For instance, autonomous vehicles traveling on highways at speeds over 100 km/h must switch from one cell to another within seconds. If the handover process is not fast enough, the vehicle may experience a temporary loss of connection which is unacceptable for mcIoT Requirement, and potentially leading to a failure in receiving real-time updates about road conditions or navigation instructions, which could compromise safety in case of emergency breaking for example.

- **Stringent QoS requirements:** mc-IoT applications require stringent QoS to ensure reliability and low latency. Disruptions during handovers can lead to delays or loss of critical data. For example, in autonomous driving, a delay in handover could mean that the vehicle does not receive timely information about an obstacle or traffic signal change, leading to a potential accident.

- **Dynamic network conditions:** During large public events, such as concerts or sports events, network congestion can dramatically increase due to the high number of users in a small area. This can interfere with the handover process for mc-IoT devices like emergency response drones or public safety vehicles, which rely on uninterrupted communication to perform critical functions.

- **Network slicing management becomes complex:** Mobility in mcIoT introduces

significant challenges for network slicing, making operational slice lifecycle management far more complex than in static slicing scenarios. This complexity arises from device mobility, strict URLLC QoS requirements, frequent handovers, as well as orchestration, automation, and isolation challenges in dynamic environments. Network slicing management can address mobility of mcIoT slices at several levels, such as:

1. *Service continuity across cells and slices:* To ensure QoS continuity during thousands of simultaneous handovers, mobility management must adress inter-slice continuity, not only cell-to-cell continuity. The handover process becomes more complex because it involves QoS-aware orchestration, rather than simple cell switching, in order to maintain seamless continuity across network slices during mobility.
2. *Dynamic slice adaptation:* When devices move across cells, the slice must be dynamically reconfigured or extended to maintain QoS, with resource allocation managed in real time for the moving devices. For instance, if hundreds of drones enter the same cell simultaneously, the slice must instantly scale up resources without violating the 1 ms latency constraint to avoid drone crashes.
3. *Cross-slice interference:* The mcIoT slice operates together with eMBB and mIoT slices; thus, resource competition (e.g., bandwidth, edge servers and theirs CPUs) needs intelligent slice orchestration. For example, during High-Speed Train (HST) travel, an eMBB slice may provide high-definition (HD) video streaming or high-speed Internet access for passengers, while a mcIoT slice for critical train communications simultaneously requires ultra-low latency and high reliability for safety and control message delivery between the train conductor and the rail operating center. Without intelligent orchestration, the mcIoT slice could suffer performance degradation due to competition with eMBB traffic. If control messages are not delivered on time, the system risks accidents or emergency stops.

- **UDN with Auto and Heterogeneous Devices & Use cases.** In mcIoT slices, ultra-dense networks host thousands of heterogeneous devices with diverse QoS needs—from simple sensors with delay tolerance to vehicles requiring real-time safety signaling. Managing UDNs with such device diversity makes mcIoT slice mobility particularly challenging, as the network must simultaneously support thousands of connections

while meeting strict QoS requirements. For example, in a train station, the slice must serve vehicls, drones, sensors, and passengers without affecting safety-critical mcIoT communications.

- **Legacy mobility management approaches are insufficient:** Legacy mobility management mechanisms enhance largely the eMBB demands for high data throughput with best-effort handovers, assuming that short disruptions are tolerable. However, they cannot address mission-critical IoT scenarios, as mcIoT requires uninterrupted service continuity, ultra-low latency, high reliability, and real-time slice orchestration under mobility. These characteristics render legacy mobility management mechanisms either unusable or insufficient, thereby highlighting the need for novel, on-demand mobility management strategies for mcMTC. For instance, Session Continuity via Anchors (SGW/UPF) is a legacy mobility function based on centralized anchoring for simplicity, where added delay is tolerable. However, it cannot be applied to mcIoT, which requires distributed and dynamic anchors to minimize latency. For example, in autonomous vehicle handovers, central detour latency is unacceptable.

While the motivation for choosing the LSTM framework as an appropriate mobility prediction method has been established, it is important to note that LSTM is particularly well-suited for addressing the challenges of mc-IoT slice management in dynamic IoT applications due to its ability to handle sequential data and capture long-term dependencies. The relevance of LSTM-based next-cell prediction can be summarized as follows:

- **Handling sequential data:** LSTMs excel at processing sequential data, making them ideal for predicting future cell connections based on historical movement patterns. This capability is essential for understanding and anticipating the mobility patterns of devices in mc-IoT applications. For instance, in connected vehicle scenarios, LSTMs can learn from past movement sequences to predict the next-cell, enabling proactive handover decisions that reduce latency and prevent dropped connections.

- **Capturing temporal dependencies:** LSTMs can capture long-term dependencies in data, allowing them to consider the temporal context of a device's movements. This results in more accurate predictions of future cell transitions. In intelligent transportation systems, LSTMs can process and learn from the temporal patterns of vehicle

movements and traffic conditions, enhancing the accuracy of next-cell predictions and ensuring seamless connectivity.

- **Adaptive and proactive resource allocation:** By accurately predicting the next-cell a device will connect to, LSTM-based models enable proactive resource allocation. This means that network resources can be prepared in advance for the incoming connection, minimizing handover latency and ensuring QoS. For UAV operations, this predictive capability ensures stable control and data links, even as UAVs move through areas with varying network coverage, maintaining the high reliability required for mission-critical tasks.

In what follows, we discuss the assumptions in mission-critical IoT slice category group as classified by the ITU organisation and what problems that arise. Then, we explain in detail the range of mobility mcMTC applications supported by the 5G network, the concept of network slicing architecture and the slice responsible for such types of applications. Further, we demonstrate how network slicing can address some of the main problems of mc-IoT applications mentioned earlier, and how the mobility prediction may assist network slicing system to handle the dynamicity of mc-IoT slices. This is establisged through a logical sequence , beginning with the critical aspects of mc-IoT communications, their applications, and the impact of mobility on this type of communication. Then, the slicing aspect is addressed through the introduction of specialized mc-IoT slices within the three-layer functional architecture of network slicing, highlighting their advantages in terms of reducing communication latency as well as managing heterogeneity and scalability. After that, we highlighting the advantages of using LSTM for historical mobility prediction and in the URLLC communication domain. Finally, we provide a detailed architectural description of the proposed LSTM model used for next-cell prediction. We explain its benefits for mc-IoT slices under network slicing. We also show how it serves as a predictive maintenance (PdM) framework that helps manage resources efficiently and reduces performance degradation during real-time critical operations. Finally, we provide a detailed architectural description of the proposed LSTM model used for next-cell prediction, by discussing its benifits for mcIoT slices under network slicing and its role as a predictive maintenance framework that supports efficient resource orchestration and mitigates performance degradation in real-time critical operations.

4.3 Mission critical IoT communication

mcMTC is intended for time-sensitive communication, which focuses on scenarios where quick, dependable responses are critical, which emphasizes high device density over ultra-low latency and reliability. This type of communication is specifically designed for mission-critical IoT scenarios operating in controlled environments. It is characterized by small payloads and low data rates, ensuring highly reliable and consistent performance for critical applications.

4.3.1 Applications, requirements and assumptions

In comparison to older cellular networks, they have been designed to satisfy the human communication application scenarios, where most efforts have focused on providing users with connectivity that accommodates the growing demand for higher data rates. In 5G, mc-IoT is a subset of IoT, designed for mission-critical IoT application scenarios that require a data delivery from end-to-end, while guaranteeing URLLC [70]. The typical mc-IoT use cases as mentioned in [32], include the communication between vehicle and the transportation infrastructure and the cooperation among vehicles in smart transportation [71, 72], remote surgery in healthcare [72], robotics cooperation or remotely maneuver in public safety agent [72] or process automation and control in industrial environments [73, 72, 74], and many others. As well, these applications are characterized by different functional and performance requirements [75], where the degree of importance of each feature differentiates from one application to another. For mcMTC services, 5G standards outline high levels of availability, while latency requirements are often set to reach millisecond-level delays or even lower [76]. The different mission-critical IoT use cases and applications with theirs diversified requirements are depicted in Figure 4.1.

Accordingly, we define mc-IoT through a set of stringent requirements and fundamental assumptions that are indispensable for supporting mission-critical applications—where even minimal disruptions or delays can result in severe consequences. The core requirements of mc-IoT encompass ultra-high reliability, ultra-low latency, and high availability. Reliability refers to the probability that a given data packet is successfully delivered within a specified time frame. It mandates that packets are not only transmitted successfully from one node to another but also that the predefined latency constraints are met. Latency is defined as

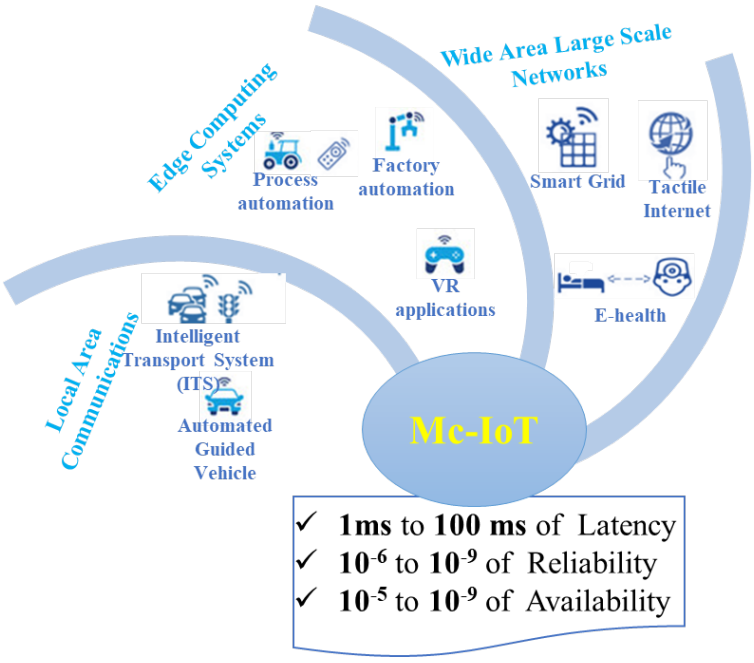

Figure 4.1: The different mc-IoT application scenarios and requirements

the time delay between the moment data is generated and transmitted by a device and the instant it is successfully received and processed by another. Finally, availability denotes the probability that a given service remains accessible, particularly in terms of coverage. For example, an availability rate of 99.99% implies that, on average, one user out of 10000 may experience insufficient coverage.

4.3.2 Mobility support

In terms of mobility requirement, mcMTC use cases may be classified from applications with high mobility support such as self-driving cars, intelligent transportation systems, railway and aviation systems, to also applications that require less or no mobility at all, such as factory automation, remote control and virtual/augmented reality. Starting from Rel-15 and continuing in the 3GPP releases, 5G focused on supporting critical MTC and enables new use cases categories by meeting the requirements imposed by URLLC transport applications, such as connected cars, high-speed trains, and unmanned aerial vehicles [77, 71]. In High mobility application scenarios, many factors will seriously degrade the system performance due to high-frequency bands, higher doppler effect, and fast movement with speeds up to 500 km/h, where the recent communication systems develop advanced innovations and technologies to provide services to high mobility MTC users at a data rate on the order of hundreds of Mbps or higher [78]. For instance, many 5G underlying advanced communica-

tion technologies such as ultra-dense networks, ultra massive multiple-input multiple-output, and millimeter wave can enhance localization, coverage, and reliability's communication performance [79, 80] for such types of high mobility MTC applications.

4.4 Network slicing architecture model

The concept of network slicing has been introduced as a key enabler to help reach the best QoS for each tenant while serving diversified and heterogeneous services and applications. As shown in Figure 4.2, the three-layered view of network slicing architecture inspired from [81] allows to provide on-demand tailored services for distinct application scenarios while using the same physical infrastructure [81]. It makes partitioning of the physical network into multiple virtual network slices, isolated from each other to build an end-to-end network service with optimal utilization of resources in terms of communication, computing, and storage [75]. The upper layer contains a variety of vertical applications supported by 5G, each application type has its own characteristics and requirements. To fulfill the requirements of each application type, 5G is classified into three slice groups: enhanced Mobile Broadband (eMBB) to support a large traffic flow cases, mcMTC slice (URLLC) for mission-critical communications, and massive MTC slice (mMTC) for massive MTC. All done through leveraging SDN and NFV technologies, which are responsible to construct, configure and manage all the network slices.

mcMTC is the most critical slice in 5G, where its main goal is to enhance the user's QoE by establishing highly reliable, low-latency, and stable communications [26]. It is intended for critical applications with URLLC and is characterized by a lower data rate, bursty traffic, and small payloads [34]. Reliability and latency are the main KPIs of the mcMTC slice. Availability, security, resiliency, and mobility are also critically important features of the mcMTC slice to provide services that are extremely sensitive to latency, such as autonomous driving [82, 81]. In addition, mcMTC is a time- and mobility-sensitive slice, which requires priority over all other mobility decisions of nodes and dedicated resources [83]. Therefore, user mobility may influence URLLC communication requirements, but the automation of network slice operations is a promising solution to improve the KPIs of mcMTC slices in a dynamic 5G communication system, requiring the design of models and operational policies

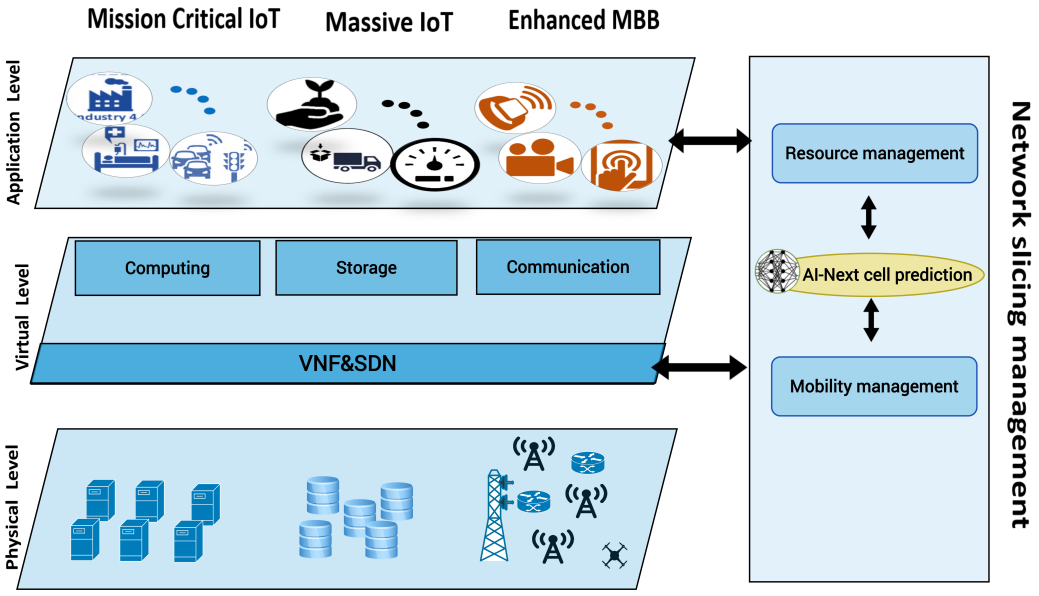

Figure 4.2: Network slicing architecture.

in an anticipatory manner.

4.5 AI-Mobility Prediction for mcIoT Slice Enhancement

Communication networks are inherently dynamic as explained in [84], which means that, in a dynamic system, when the network parameters and/or topology change frequently, the network slicing management will require proactive decision-making to adjust network slices to match the network changes. Furthermore, prediction techniques are relevant solutions that can be used for dynamic network slicing and the automation of 5G network slicing functions [84, 85]. The basic idea is when the system will be able to predict the behavior of users' mobility and the status of the network slices parameters (such in [67] and [23, 86, 87] respectively), then the predicted information can be employed to make decisions to update the system accordingly, to better preserve resources and maintain connections stability with users of slices. On another hand, we focus on AI mobility prediction engines, because AI-based solutions thrive particularly in the context of MTC due to its heterogeneous requirements [30]. In particular, providing AI-movement predictability in mcMTC networks is crucial to preserve resources, maintain connection continuity, and meet their critical QoS

requirements. Hence, supporting URLLC applications with high mobility in wireless communication systems will bring significant research challenges [80], especially at the virtual level of network slicing systems.

Within the scope of cellular networks mobility prediction is considered one of the main investigation fields, which is used to optimize the performance of various network functions such as mobility management, resource management, and location-based service [88]. History-based mobility prediction is one way to address the influences of user's movement, by estimating their user mobility patterns based on their movement history records, and therefore be able to predict the future movement of a given actual state [88]. There are different kinds of mobility prediction methods based on historical movement data, but through accuracy evaluation studies on location and trajectory predictions [52, 89] data-driven methods (e.g., ML) can reach better prediction accuracy compared with model-based methods, and can be applied to further improve the performance for data with short and long-term dependencies. Moreover, the particular ML supervised deep learning algorithms are very powerful when the amounts of available history data are extremely large, which will be able to find the optimal solution in near real-time [89] so as to make regression or classification tasks. Meanwhile, the historical movement resulting from prior user movement trajectories may be approached as a time series of locations, where the Deep Learning LSTM algorithm is very powerful for time-series prediction as shown in [90, 91], where authors have proved through a set of experiments that the Recurrent Neural Network LSTM outperforms statistical model ARIMA and traditional ML Support Vector Machine model SVM on time series forecasting in both traffic and user mobility prediction. Besides, the mobility prediction model based on LSTM can extract hidden correlations in the user's movement history to make predictions with input sequences of different sizes (at short-term or long-term levels).

LSTM has been applied in various fields in computer science, not only in the natural language processing field (i.e. machine translation [92] and speech recognition [93]) but also adopted in wireless networks fields and improved the KPIs of communication systems. Within the framework of URLLC communication systems as presented in [89], LSTM has shown its superiority in different time series problems; such as channel prediction [94, 95], mobility prediction [96, 97], traffic prediction [26], and also QoS/QoE prediction [98]. However, only a few works have investigated LSTM under network slicing architecture. Li et al. [67] proposed an intelligent resource management solution for RAN slicing that may cap-

ture the user mobility and enhance the system utility, while Abbas et al. [23] proposed a traffic LSTM-based solution toward the automation of network slice life-cycle management and resource scalability assurance at the operational level, and the authors in [63] have proposed a dynamic bandwidth resource allocation method for transport network slicing based on traffic prediction using LSTM model. As a result, due to the availability of large-scale mobility profiles in the network captured from the mobility of their connected URLLC users, a supervised deep-learning LSTM is a candidate solution for the mobility prediction problem that may be applied to handle the dynamicity of URLLC networks in network slicing systems.

4.6 Next-Cell awareness in 5G network slicing system

In 5G communication systems, positioning plays a major role to provide accurate location information in both indoor and outdoor environments, which can be utilized not only for location-based services but also for improving wireless communication performance like routing and network optimization, and a vast set of location-aware radio resource management (RRM) functionalities [99, 100]. For instance, the enhanced Cell-ID method (E-CID) is a positioning method already available in rel9, it is a cellular-based localization method typically applicable in urban and indoor environments when there is a lack of satellite visibility, E-CID is an enhanced version of Cell-ID based methods in which cell-identity information is combined with other measurements to improve the positioning accuracy capabilities [99], and therefore cell-identity and other measurement-related information of passing users are captured at the network side and thereafter they will be available at the network in an updated and reliable manner.

From one side, location information can benefit the performance of the 5G communication systems to rapidly control their operations and improve their KPIs requirements through designing communication systems in an anticipatory manner [101, 100], and from another side, the location in terms of the Cell Identity is the key mobility-related factor in the cellular network, as the mobility management definition in [102] that is the process in which the network is able to know the cell that is currently in use by a user. For such reasons, to improve the network slicing systems to fulfill the desired URLLC networks requirements in a dynamic environment, next cell prediction is one way to tackle these issues, that will

address the geographic movement effect of their connected MT-users at the virtual level, in order to provide proactive control and management over URLLC network slices.

Therefore, in mcMTC applications of medium and high mobility scenarios, when the MT-user is in motion, the serving cell will be changed over time which causes more frequent handover, and the ability of a service provider (SP) to predict the cell that will be traversed in the near future may perform seamless mobility and fast handover control, wherein the network slicing management entity will be aware enough of the toward cell that will be visited to perform the suitable updates for network slice, so as by preparing mobility and resource management algorithms required in advance before the handover occurs. Thereafter, Next-cell awareness can be harnessed in 5G network slicing architecture for the following aspects:

1. In mcMTC slices, the current serving cell information is also reliably and availably ensured in the network because the Cell-Identity is a network based information. Also the latency of URLLC communications will be reduced, because as same as all dedicated functions of critical slice the AI next cell prediction module should be deployed at the edge cloud for fast decision making.

2. We find such types of applications in urban and indoor environments, where the 5G network becomes ultra-dense due to the dense distribution of small cells within a macro cell, and a redundant coverage is established which can be used to ensure the reliability for URLLC slices based on the predicted ID_{Cell} information. That is why the predicted ID_{Cell} information may optimize the handover operation between neighboring cells, by a prior update of the Neighbor relation Table (NRT) of eNodeB when a moving cell or a newly added small cell are detected.

3. When the MT devices are continuously moving from one cell to another, it is necessary to perform corrective actions in advance to handle the mobility over network slices. Thus, to avoid service degradation, and resource wastage into critical slices, AI-next cell prediction algorithm can benefit both resource management and mobility management planes for the 5G network slicing architecture. This means that the predicted next cell information is used as a movement indicator to run the dedicated mobility and resource management algorithms of critical slices.

4. Automatic processing of network control and management is one of ITU-T's key goals

in the IMT-2020 network [27], which provides a dynamic network slicing solution in 5G systems to address users' varying communication requirements and to match the network changes. Therefore, based on our next cell prediction module, the network slice management will be extended toward proactive and dynamic update support of network slices, so that there will be critical slice-aware mobility. Accordingly, it may be applied to allow the automation of mcMTC slice operations, including the automation of the life-cycle management of critical slices to react to dynamic network changes in near real-time.

5. Self Organized Networks (SONs) play also a major role in the automation of 5G network slicing management, and Automatic Neighbour Relation (ANR) is a SON function that automates the management of adjacent neighbours in the eNBs, but not on a real-time basis [103]. In critical applications having high mobility support, the neighbour cell lists are frequently modified, which requires the ANR function to be able to optimize neighbour cell lists more quickly. This can be effected based on the prior knowledge of the cell that the moving user will connect to in the near future, which can extend the capacity of the ANR function to respond to the change of network topology for a proactive resource adjustment and therefore achieve a timely hand-off.

6. To realize IoT with seamless and ubiquitous computing, energy efficiency must be improved due to constrained resources and limited power capabilities. This is particularly critical in mcIoT systems, due their strict latency requirements and uninterrupted connectivity demands. Generally, the energy aspect refers to: 1)Preserve the battery life of sensors and machine-type devices such as vehicles and drones. 2)Reduce the network's energy costs by minimizing excess energy consumption from antennas, base stations, and servers. Network slicing, combined with NFV and SDN, plays a key role in optimizing the energy aspect while guaranteeing safety-critical QoS. When integrated with next-cell prediction, energy efficiency is further improved by enabling the network to anticipate future cells, prepare handovers in advance, and avoid unnecessary signaling and redundant handovers.

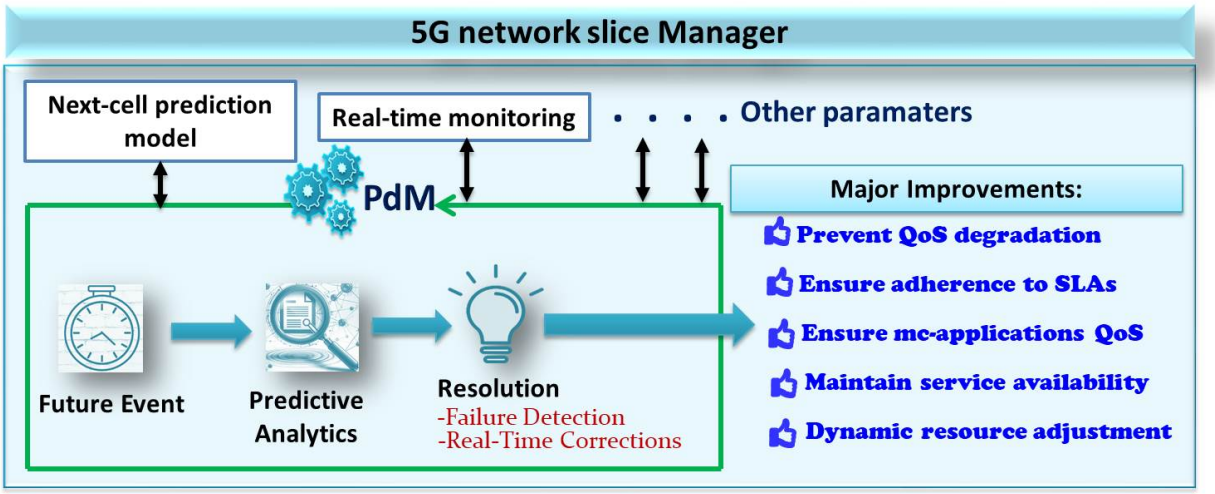

Figure 4.3: PdM strategies for enhancing 5G mcIoT slice QoS.

4.6.1 Next-Cell Prediction for Slice Maintenance under Mobility

The proposed next cell prediction architecture acts as a predictive maintenance framework (PdM) within 5G-based network slicing systems. The deployed next-cell model enables the network slicer to determine in which cell the user is located in near real-time based on its current serving cell. This helps PdM to maintain the network slices proactively, thereby directly improving the reliability and QoS of mission-critical IoT services and mitigating disruptions under mobility.

This is illustrated in Figure 4.3, where PdM in 5G networks functions as a modern SoN system for the network slice manager to enhance and preserve QoS. PdM leverages data monitoring, AI tools, and mobility prediction analytics to proactively prevent network component failures and implement timely interventions to maintain consistent network functionality.

Moreover, PdM operates in a closed-loop automation process :

1. **Monitoring :** The system continuously collects real-time data and mobility predictions.

2. **Detection :** Events are triggered only when potential failure signs are detected.

3. **Analysis :** PdM applies predictive analytics on the data to anticipate potential constraints before they cause service degradation.

4. **Resolution Actions :** corrective measures are proactively triggered to prevent failures and maintain slice service quality in a timely manner. Different forms of corrective actions are applied to uphold consistent network functionality.

4.7 The proposed solution

In this section, we present our mobility prediction framework based on Long Short Term Memory (LSTM). Our solution is a history-based mobility prediction, which exploits the trajectories of a 5G moving user to predict the future location that will be visited. Instead of geographic location, we talk about location in terms of the Cell Identity (ID_{Cell}), which is corresponding to the base station that serves the user. The proposed prediction model leverages the trajectories of users for training the LSTM deep neural network for the prediction of the future serving cell for a given user. That will bring a significant improvement in the 5G communication systems. Hence, if the network is able to identify the most likely future cell of MT-users based on their identifiers, this will assist the network slice management in enabling sensitive slices (i.e., mcMTC slices) to adapt to the network dynamicity by triggering the required optimisation operations in advance.

We have chosen LSTM because it has powerful ability to predict time series with short and long-term dependencies, it works better when the historical data are large, and also because it has the capabilities to remember inputs for a long time, thereby the next location can be predicted more accurately. The rest of this section is organized as follows: the notations used in this paper are listed in Table 4.1, followed by the problem definition and some LSTM preliminaries, then by a detailed description of our proposed LSTM architecture for the next cell prediction.

Table 4.1: Notations

Notation	Definition
p_i^u	The location ID_{Cell} of a moving user u at the time instance i
$\tilde{p}_i^u$	The predicted user's u new location ID_{Cell}
$Tr^u = (p_1^u, p_2^u, \cdots, p_m^u)$	A trajectory of a moving user u with length m
$Tr^{u'} = (p_1^u, p_2^u, \cdots, p_k^u)$	The input trajectory of length k
Δ	The time step length
B	The set $\{0, 1\}$
P	The probability value is a real value $\in [0, 1]$

4.7.1 Problem definition

We define the mobility prediction problem as a multi-class classification problem. The trajectory can be approached as a time series of locations, and defined as a sequence of time-

stamped locations that are chronologically visited. A location will be represented by the crossing cell traversed by the user u, and a timestamp t.

Formally a location p is the pair $p_t^u = (ID_{Cell}, t)$, where ID_{Cell} is the Cell Identifier of the visited cell by a moving user u at a timestamp t. Hence, the trajectory of the user u, Tr^u is written as $Tr^u = (p_1^u, p_2^u, \cdots, p_m^u)$, where $m = \mid Tr^u \mid$ is the trajectory length. The time interval between each two successive locations is equal to $(t_i - t_{i-1} = \Delta$, where $1 \leq i \leq m)$. Let $Tr^{u'} = (p_1^u, p_2^u, \ldots, p_k^u)$ be the actual input trajectory, where p_k^u is the current serving cell ID_{Cell} of the user u at the current time instant k, we seek to predict the future cell that will be visited in the next time step $(k+1)$ that maximize the probability condition of the target location p_{k+1}^u given a trajectory source $Tr^{u'}$ such as

$$\tilde{p}_{k+1}^u = \arg\max_{p_{k+1}^u} P(p_{k+1}^u \mid Tr^{u'}). \tag{4.1}$$

The performance metrics of the model will be evaluated using data through multi-class classification tasks with the following criteria: accuracy, precision, recall, F1-Score, and categorical cross-entropy. Accuracy measures the proportion of correct predictions among the total predictions, providing an overall effectiveness of the model. Precision evaluates the proportion of true positive predictions among all instances classified as positive, indicating how many of the predicted positive instances are actually correct. Recall, also known as sensitivity or true positive rate, measures the proportion of true positive predictions among all actual positive instances, reflecting the model's ability to capture all relevant cases. Finally, F1-Score is the harmonic mean of precision and recall, providing a balanced metric that accounts for both false positives and false negatives, especially useful when dealing with imbalanced datasets. Detailed definitions and mathematical formulations of these metrics are given in Section 5.3.2.

4.7.2 LSTM Preliminaries

LSTMs [104] are improved RNNs that have been designed to address their limitations, by carrying on long-term dependencies in relation to the gated structure. LSTMs are able to remember information for long periods of time through a gating mechanism to avoid the RNN vanishing gradient problem [104]. Moreover, LSTM architecture is a black-box framework composed of a series of connected units called memory blocks. Each LSTM block consists

of multiple memory cells with self-connections storing capabilities of the temporal state of the network. Each memory cell has multiplicative units called gates which control the flow of information [105].

Moreover, as expressed in [106], LSTM is defined as a Supervised DNN, where its goal is to learn a function (model) f that maps input sequence $X = (x_1, x_2, \ldots, x_T)$ to output Y for a period of time T, where $Y = f(X)$. The optimal model f^* is obtained after T iterations (from t = 1 to T) of the recursive formulas (4.2). The following formulas describe how the LSTM works within each memory block at a given time step t:

$$\begin{aligned} f_t &= \sigma(W_f.[h_{t-1}, x_t] + b_f); \\ i_t &= \sigma(W_i.[h_{t-1}, x_t] + b_i); \\ \tilde{C}_t &= tanh(W_C.[h_{t-1}, x_t] + b_C); \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t; \\ o_t &= \sigma(W_o.[h_{t-1}, x_t] + b_o); \\ h_t &= o_t * tanh(C_t), \end{aligned} \tag{4.2}$$

where f_t, i_t, o_t, C_t are respectively forget gate, input gate, output gate, and memory cell vectors, W_f, W_i, W_o, W_C and b_f, b_i, b_o, b_C are the corresponding weighted matrices and bias factors of LSTM. x_t and h_t represent the input and the output at the current time step t, while h_{t-1} is the output of the precedent time step, σ and $tanh$ are the sigmoid and the hyperbolic tangent Activation Functions respectively, and (*) denotes the element-wise product.

The author in [107], describes in detail the role of each equation (i.e., from formulas (4.2)) of a recursive module of LSTM. The three gating vectors f_t, i_t, and o_t are responsible to decide which values shall be forgotten, updated, and preserved from the cell state respectively, using the sigmoid layer of each gate. Whilst, the tanh layer creates a vector $\tilde{C}_t$ of new values that could be added to the state. The new cell state C_t is obtained based on the previous cell state, the input and forget gate decisions, and the vector of new candidate values $\tilde{C}_t$. In the end, the final LSTM output h_t will be the filtered version of the current cell state C_t.

A Deep or a Stacked architecture has been proven to be powerful in representation

learning [108, 109]. For instance, a stacked LSTM optimizes the power of the neural network, by allowing it to make better utilization of features by dispatching them over the space via multiple layers. So, it is used to extract the high-level representation and encode temporal correlations on robust local features [110, 111]. Moreover, a stacked LSTM stacks multiple LSTM layers, in which each LSTM layer's output work is an input to the next LSTM layer [112]. Hence, the first LSTM layer is fed with the input sequence, the second layer is fed with the output of the first layer, and so on.

Therefore, at a given time step t, each LSTM cell of the hidden layer $(L-1)$ outputs a sequence h_t^L that will be used as input to the upper hidden layer L. It can be calculated using equation (4.3) as follows

$$h_t^L = LSTM(h_t^{L-1}, h_{t-1}^L, C_{t-1}^L), \tag{4.3}$$

where LSTM(.) is the mapping function calculated by the equations of formula (4.2) that update the LSTM cell parameters at instance t. Hence, h_t^{L-1} is the hidden state vector of the lower hidden layer $L-1$ at instance t. C_{t-1}^L and h_{t-1}^L are the memory cell and hidden states of the hidden layer L at time step $t-1$.

In the end, the output result of the last L^{th} LSTM layer will be the vector $h^L = (h_1, h_2, \ldots, h_T)$, fed as input to the fully connected output layer to extract the predicted information Y that maximizes the conditional probability of Y given a source X (i.e., $\arg\max_Y P(Y \mid X)$).

4.7.3 LSTM Framework for next cell prediction

The proposed predictor has two aspects: 1) it is a multi-class classification task, 2) it also belongs to the spatial time-series forecasting problems class. The first aspect is justified by the fact that the output y takes one discrete value as a specific class label; which is the corresponding ID_{Cell} that will be visited by the user u in the near future. As for the second aspect, the prediction task is approached as a spatial time-series since input data x are represented by a sequence of the locations (ID_{Cell}) orderly visited by user u, in which there is an auto-correlation between the cells that have been visited in the past and the user's next cell that will be visited in the future.

Furthermore, based on the preprocessed dataset of the raw mobility traces of 5G users,

the proposed next cell prediction framework intends to learn the users' mobility behavior in order to extract patterns hidden in their historical motion data. More precisely, the decision-making of the proposed framework is based on many-to-one scenarios with multiple inputs-single output. In which, the solution of our model consists of inferring the most likely next-cell location of a specific user in the next timestamp, given a sequence of cell locations previously visited by that user.

Figure 4.4 shows the proposed framework for next-cell prediction. Our architecture is mainly inspired by the work of Hua et al. [91] who have worked on time series prediction using RCLSTM (a variant of LSTM where neurons are randomly connected) whereas our approach uses a fully connected neural network. The following sections outline the details of

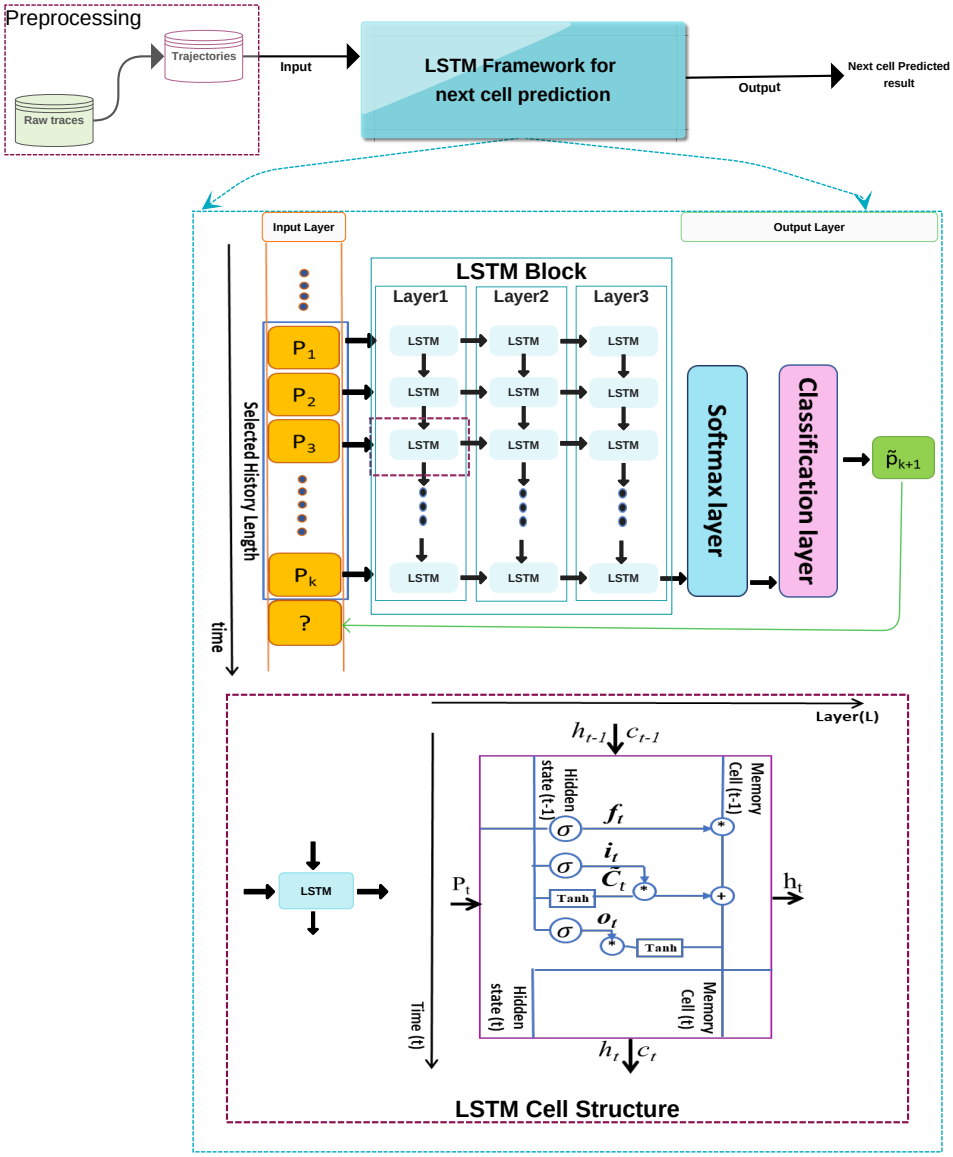

Figure 4.4: Our proposed next-cell classifier.

the proposed framework, from the preprocessing to the deep learning modeling-based LSTM phase.

4.7.3.1 The preprocessing phase

In the preprocessing phase, we have applied data exploration and transformation techniques on the users' movement history dataset. First, we transform the initial dataset (raw traces) into discrete trajectories of ordered visited locations (ID_{Cell}) as shown in Figure 4.4. Thus, the mobility traces of every moving user u is transformed into a set of trajectories, and each trajectory of a moving user u is represented by a set of sequence of locations orderly visited in the past. Hence, for a given user u, the trajectory result Tr^u of length m is formed by an ordered set of sequences with equal history length $k+1$. Each sequence x_i is represented by a vector $x_i = (p_i^u, p_{i+1}^u, \cdots, p_{i+k}^u)$, and each location p_t^u has two attributes $p_t^u = (ID_{Cell}(t), ID_{User}(t))$, with (ID_{Cell}) representing the cell identity of the serving cell that serves the user u at the timestamp t, attached to the user identifier (ID_{User}).

Second, as a multi-classification modeling task, and to feed the deep LSTM neural network with appropriate input and output data (X and y). The dataset of trajectories are arranged in a features matrix $X \in R^{n \times k \times 2}$, and a target vector $y \in B^{n \times N}$ of labels, with N being the number of labels. Thus, each entry x_i from X features two attributes: the user identifier (ID_{User}) and the k past visited cells $x_i = (p_i^u, p_{i+1}^u, \cdots, p_k^u)$, and each output y_i from y indicates the tuple representation of the last $(k+1)^{\text{th}}$ value (i.e., p_{k+1}^u) of each row from the dataset.

4.7.3.2 The classifier framework description

The next-cell prediction model built by using LSTM algorithm, as shown in Figure 4.4, has the advantage of stably capturing the long-term dependency of real mobility patterns of MT-users from their historical trajectories. This is owing, as explained in Section 4.2, to the collaboration between the gates and the memory cell of each LSTM cell.

For the next cell prediction modeling, the set of the pre-processed trajectories from raw traces of multiple users is used as input data to the LSTM deep learning framework. It is made in the form of samples from $X = \{x_1, x_2, \cdots, x_n\}$ and the set of outputs from $y = \{y_1, y_2, \cdots, y_n\}$. Let $\tilde{y} = \{\tilde{y_1}, \tilde{y_2}, \cdots, \tilde{y_n}\}$ denotes the set of serving cell that we aim to

forecast. Within each period of time k, given a short mobility trace $x_i = (p_1^u, p_2^u, \cdots, p_k^u)$ from X. Each LSTM cell executes the equations of formula (4.2) in an iterative and in an ordered manner, in order to predict the future cell that will be visited ($\tilde{y_i} = \tilde{p}_{k+1}^u$) at the next timestamp $k+1$.

The predictor architecture (cf., Figure 4.4) is LSTM-based next cell prediction framework. The developed deep architecture with three stacked layers gives better results compared to single-layer LSTM, because it has more capabilities to extract the high-level representation and to encode temporal correlations on strong local features. It is composed of three major components correspond to three main actions involved in the prediction task. First, in the input layer, input data are sequences of size k with two-dimensional input feature. This layer treats each given sub-trajectory $x_i = (p_1^u, p_2^u, \cdots, p_k^u)$ to feed the first layer of LSTM block of next layer.

Then, we have the second and the output layers which are the main parts of the architecture. They leverage the deep learning LSTM tools to perform our mobility prediction task. More specifically, the second layer is responsible for the sequence processing coming from the input layer. It is formed by an LSTM block which includes 3-stacked LSTM layers, each with 64 hidden units (this is the optimal dimension that we have obtained during the hyperparameter tuning process when we trained the model). Each LSTM layer's output is an input to the next layer. In other words, the first LSTM layer is fed with the input sequence of the input layer, the second layer is fed with the output of the first layer, and so on.

Finally, based on the output vector of the previous layer, the output layer is used to explicitly reveal the predicted next cell location ($\tilde{p}_{k+1}^u$). It is composed of two layers: 1) the softmax layer, and 2) the classification layer. The softmax layer is a fully connected neural network, formed by N units to specify the number of classes of the classifier, and followed by the softmax activation function. It receives the final trajectory vector from the previous layer $v = h^L = (h_1, h_2, \cdots, h_k)$, where each hidden state $h_i \in R^H$ has a size H equal to the number of hidden units of LSTM layer. This vector is then mapped to the pre-activation (logit) vector $z \in R^N$, and the softmax function (e.g., as defined in Eq: 4.4) is applied to z to produce the output vector $\tilde{y}$, which represent the probability distribution over the N

class labels, given by $\tilde{y} = (P(y = c_1 \| v), P(y = c_2 \| v), \cdots, P(y = c_N \| v))$.

$$\text{Softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{N} \exp(z_j)}. \tag{4.4}$$

At the end, the classification layer is used for final cell location mapping. Based on the probability distribution output y' received from the softmax layer, the predicted next cell location ($\tilde{p}^u_{k+1}$) is obtained By applying the maximum likelihood method using the argmax function, which maps the equivalent class label having the highest probability in $\tilde{y}$.

Moreover, the cost function we optimize is the categorical cross-entropy loss function $L_{CE}(y, \tilde{y})$, defined for all inputs to calculate the total loss error between the LSTM-predicted data $\tilde{y} \in P^{n \times N}$ and the ground truth data $y \in B^{n \times N}$, thus guiding the optimization process during training to find the optimal model f^* with minimal cost. In a probabilistic aspect, it is a probabilistic metric and a generalization of the cross entropy loss function of Shannon entropy [113] used in the binary classification task. That measures the deviation of the learned mapping $\tilde{y}$ from the ground truth data y. The loss function L_{CE} is calculated using formula (4.5) such as

$$L_{CE}(y, \tilde{y}) = -\sum_{i=1}^{n} \sum_{j=1}^{N} y_{ij} log(\tilde{y_{ij}}), \quad \text{for n samples and N classes,} \tag{4.5}$$

where y_{ij} is the true one hot encoder vector of sample i, and $\tilde{y_{ij}}$ being the predicted probability vector of sample i being of class j. $y \in B^{M \times N}$ and $\tilde{y} \in P^{M \times N}$

4.7.3.3 The training and prediction phases

On completion of the pre-processing phase and the architectural detail description phase of the proposed next-cell classifier, the model execution phase starts in order to find the one that best fits the data in the dataset.

As a supervised learning task, the execution phase operates in two phases namely, the training phase and the prediction phase. Thereby, we split the dataset of trajectories prepared in the pre-processing phase into two instances *D-train* and *D-test*. *D-train* is used to train the LSTM model and *D-test* is used to evaluate its performance (cf., Figure 4.5).

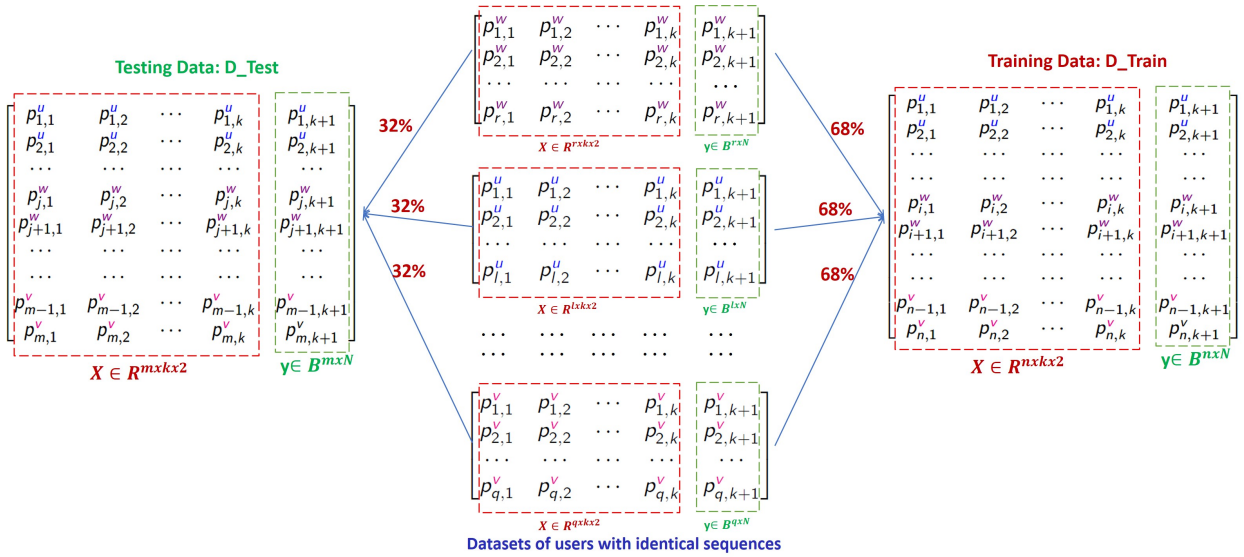

Figure 4.5: Dataset split into training and testing sets.

Besides, for an accurate prediction results, we take the larger portion of the samples (x_i, y_i) from *D-train*, corresponding to about two-thirds of the entire dataset, to train the LSTM neural network. Whilst, the remainder (i.e., samples (x_i, y_i) from *D-test*) is reserved to evaluate the trained model.

Training phase The training process' goal is to minimize the total error of the prediction using the gradient descent learning algorithm in combination with the backpropagation algorithm (BPTT) [114]. Therefore, during the training phase, the algorithm updates the LSTM parameters (weights and biases) time by time to find the optimal model f^* that makes the loss cost function as small as possible, and therefore the log-liklihood objectif function is

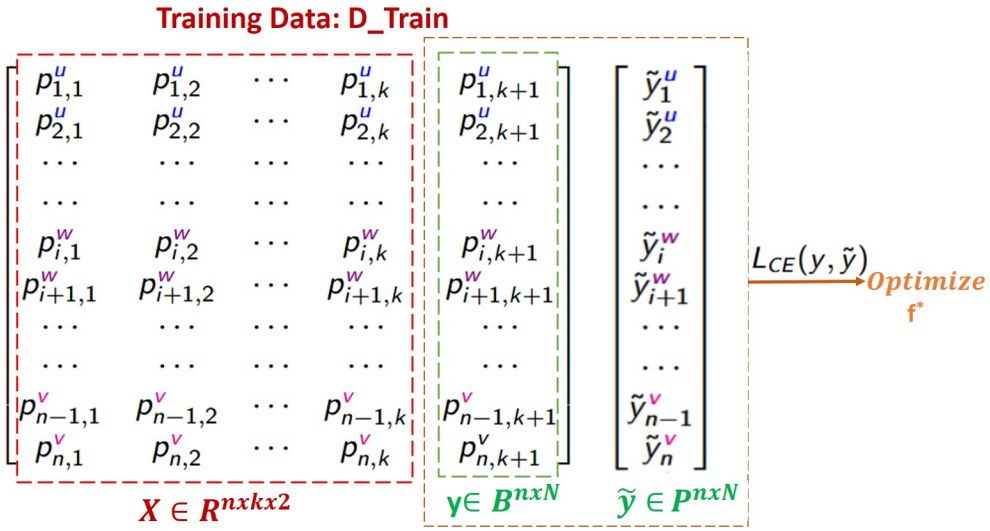

Figure 4.6: Training phase to find f^* for a given iteration.

maximized.

Moreover, the LSTM algorithm use as input the matrix of samples (X) and the binary matrix of labels (y) of *D-train* to train the model. Hence, due to LSTMs being able to remember their inputs information for an extended period of time. For each period of time k, we use as input the whole sub-trajectory $Tr^{\tilde{u}}$ of length k except the last location ($x_i = (p_1^u, p_2^u, \cdots, p_k^u) \in X$), to forecast time series to predict the next location ($\tilde{y}_i = \tilde{p}_{k+1}^u$), where the ground truth label ($y_i = p_{k+1}^u \in y$) is used to estimate the loss error L_{CE} of prediction.

As depicted in the Figure 4.6, the categorical cross-entropy loss L_{CE} using the equation (4.5) is computed at each iteration until the optimal model f^* is obtained. The proposed next-cell classifier uses an LSTM based on the mini-batch technique for faster convergence and improved accuracy. By using mini-batch gradient descent, this computation is repeated for every batch. The final epoch yields f^* with the best parameters W_i and b_i. The reported loss corresponds to the average over all batch losses.

Finally, when the optimal model f^* is generated in the training phase, a separate instance *D-test* will be used in the prediction phase to measure its performance.

Prediction phase When the training phase is terminated, we move on to the prediction phase. The latter represents the testing of the generated model f^* of the training phase. In

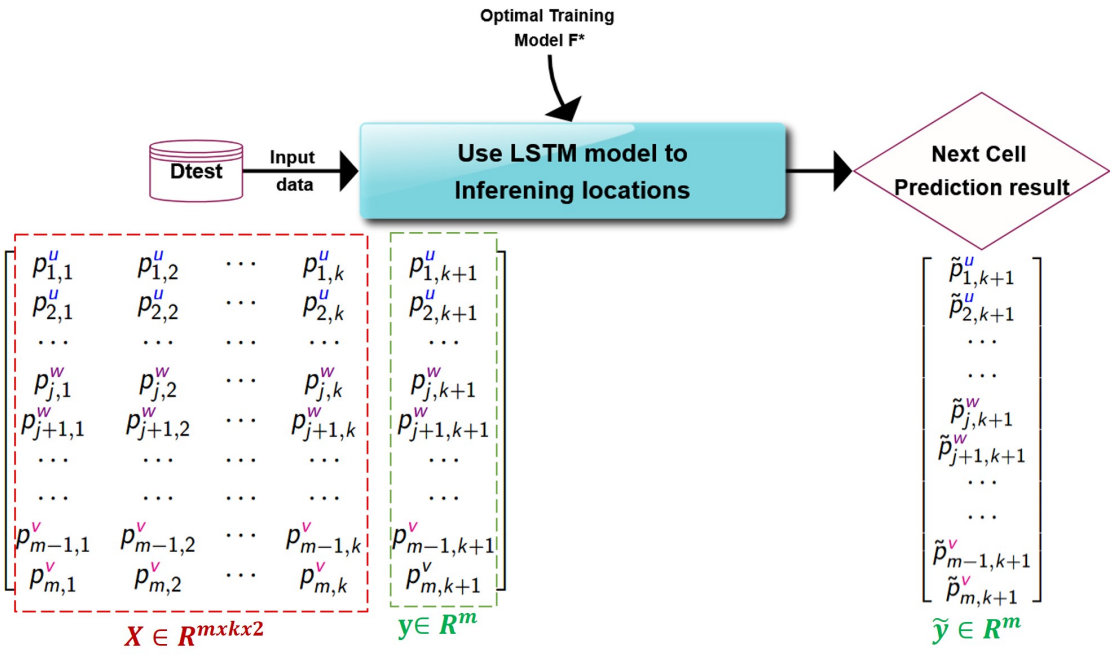

Figure 4.7: Testing phase for the next-cell prediction.

this phase, we consider that all the inputs are from *D-test* dataset (cf., Figure 4.7). Wherein, the LSTM prediction algorithm use each sequence for a given user u ($x_i = (p_1^u, p_2^u, \cdots, p_{k-1}^u)$) from the *D-test*, as an input and predict the location ($y_i = \tilde{p}_k^u$) as an output (prediction data) in the time slot k while using the parameters of the generated optimal model f^*.

Ultimately, the model is evaluated by calculating the classification error (or accuracy) metric, which depends on the percentage of miss-classified (or well-classified) prediction data in comparison with ground truth data.

4.8 Conclusion

In this chapter, we focus on AI-based mobility prediction mechanisms within the scope of high mobility mcMTC communication applications scenarios. We have shown that LSTM DNN architecture is well suited for mobility prediction, particularly the stacked LSTM architecture which is known to further empowers the neural network. Therefore, we have proposed a mobility prediction framework (i.e., an AI next-cell prediction framework) based on the stacked LSTM architecture and the trajectories resulting from the collected mobility traces of MT users. The basic idea is to predict the user's next-cell to make proactive decisions accordingly, that will optimize the KPIs of URLLC communication systems in a dynamic environment. In the beginning, we presented the network slicing architecture and our mobility prediction module that play a pivotal role in handling the dynamicity of mcMTC slices and fulfilling the URLLC QoS requirements. Then, we defined the mobility prediction problem as a multi-classification task, followed by an extensive description of our proposed framework for the next-cell prediction.

Chapter 5

Performances Evaluation

Contents

5.1 Introduction 60

5.2 Dataset generation 61

5.3 Experiment settings 64

5.3.1 Datasets splitting for training and evaluation 64

5.3.2 Evaluation metrics 64

5.3.3 Models hyper-parameters tuning 66

5.3.4 Results and analysis for each dataset 67

5.3.5 Results and analysis using multi-classification metrics 76

5.4 Practical integration of the proposed approach 80

5.4.1 Scalability concerns 80

5.4.2 Computational requirements 81

5.4.3 Potential challenges 81

5.5 Conclusion 82

5.1 Introduction

In this chapter, we analyze the efficiency of the proposed next-cell framework within the scope of URLLC vehicle communication. We consider a vehicular urban environment case study as an example type of mcMTC use cases with high mobility scenarios. First we describe

the generated dataset, which are a real vehicle mobility dataset generated through SUMO. Then, we present the set of experiments have been conducted on the proposed classifier using the datasets with various history lengths. Further, for a comprehensive analysis, we compare the proposed LSTM-based predictions with baseline ML methods: the K-nearest neighbor (KNN) algorithm, the Random Forest (RF) algorithm, and the Support Vector Machine (SVM) algorithm. At the end, we highlighted the practical implications of integrating the proposed LSTM-based approach into real-world 5G mc-IoT deployments. by addressing scalability concerns, computational requirements, and potential challenges.

5.2 Dataset generation

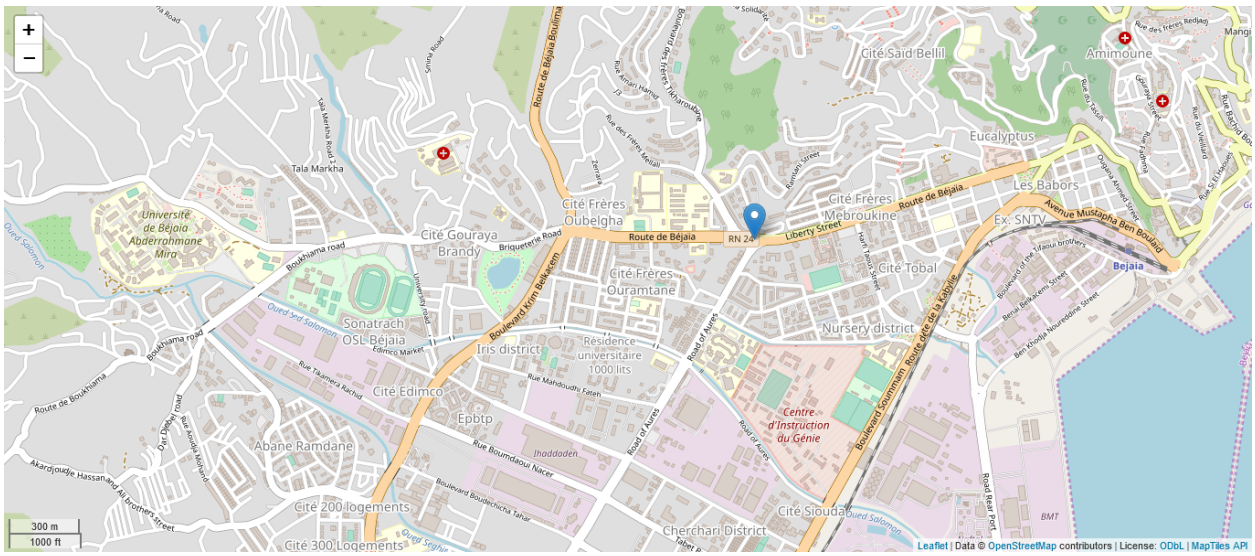

Figure 5.1: Area of the mobility simulation.

To evaluate the performance of our mobility prediction framework, we performed a set of simulations on datasets that we generated [115]. We considered the mobility traces generated using SUMO (Simulation of Urban MObility) [10] in an urban area of 5000m × 5000m of the city of Bejaia derived from OpenStreetMap [116] and shown in Figure 5.1. The output traces of the mobility generator SUMO provide detailed vehicle mobility data collected at regular intervals. The data includes GPS positions, which provide precise latitude and longitude coordinates of the vehicles at each timestamp. The vehicles' GPS positions are sampled every $1 - 5$ seconds, allowing for detailed capture of vehicle movements and thus ensuring high temporal granularity. The vehicles move freely at averagely 45 km/h over a period of 1h30 in a city center with traffic lights, frequent intersections, and dense traffic conditions.

	Profile	x_coordinate	y_coordinate	Cell_ID
0	1	5.082914	36.752655	RSU35
1	1	5.082898	36.752664	RSU35
2	1	5.082871	36.752683	RSU35
3	1	5.082831	36.752722	RSU35
4	1	5.082775	36.752779	RSU35
...	...	...	...	...
3429	1	5.082734	36.752820	RSU35
3430	1	5.082658	36.752900	RSU35
3431	1	5.082569	36.752997	RSU35
3432	1	5.082472	36.753104	RSU35
3433	1	5.082367	36.753221	NaN

(a) Profile of user u_1.

	Profile	x_coordinate	y_coordinate	Cell_ID
0	2	5.071922	36.747954	RSU29
1	2	5.071907	36.747960	RSU29
2	2	5.071876	36.747973	RSU29
3	2	5.071828	36.747993	RSU29
4	2	5.071762	36.748020	RSU29
...	...	...	...	...
5144	2	5.070905	36.748406	RSU29
5145	2	5.070819	36.748485	RSU29
5146	2	5.070736	36.748565	RSU29
5147	2	5.070659	36.748640	RSU29
5148	2	5.070576	36.748719	NaN

(b) Profile of user u_2.

Figure 5.2: Collected mobility information profiles.

The dataset also considers 42 cell base stations, known as Road-Side Units (RSUs), being deployed at fixed locations in the simulation area. These RSUs, with a coverage area of 500 meters, are capable of providing on-demand services to moving users (i.e., vehicles) passing through them. Based on this initial dataset, we generated a mobility profile for two users u_1 and u_2 with three randomly selected destinations (e.g., work-supermarket-home). These profiles describe random mobility around these destinations and include 10 different trajectories with a total number of 3434 and 5149 locations, respectively. We then carried out a mapping of each GPS position in these trajectories to the fixed positions of the RSUs deployed in the simulation area, to identify the RSUs that users u_1 and u_2 passed through. Figure 5.2 present snapshots of the collected mobility information records for each profile, including GPS coordinates and cell identifiers.

Finally, we have built k datasets ($k = 1, \cdots, 6$) as shown in Figure 5.3, where each entry has two attributes $\langle ID_{User}, k - Past - ID - Cells \rangle$, each denotes the identifier of a moving user u and and its previously visited RSUs with a k-length history $(p_1^u, p_2^u, \cdots, p_k^u)$, respectively. For instance, the cell identifier $RSU35$ is mapped to 35, $RSU2$ to 2, and so forth. The same mapping is applied to the two user profiles, where profile u_1 corresponds to 1 and profile u_2 corresponds to 2. These datasets are then used as inputs to our LSTM classifier to predict the next cell that will be visited by a user u.

By considering vehicle mobility data collected in an urban area, the generated dataset realistically reflects the frequency of inter-cell handovers encountered in real-world mc-IoT

networks. Indeed, urban environments typically have a high density of cell base stations to support high data demand as vehicles move through the city. Additionally, they present complex mobility patterns (e.g., variable speeds and directions, high-volume traffic) and obstructions (e.g., buildings, tunnels), which can stress-test the handover mechanisms as well as the capacity and robustness of the network, both of which are essential for the reliability and performance of mc-IoT applications, such as in smart city deployments. Hence, these characteristics offer a rich and challenging set of conditions, which enable the proposed LSTM-based prediction model to simulate the handover dynamics encountered in mc-IoT networks, ensuring that the model can be effective in real-world scenarios.

Profil	P_0	P_1
.....	.....	.....
1	25	15
1	15	15
.....	.....	.....
1	3	3
1	3	2
.....	.....	.....
2	3	3
2	3	8
.....	.....	.....
2	31	31
2	31	29
.....	.....	.....

(a) dataset_1 with one sliding window length.

Profil	P_0	P_1	P_2
.....	.....	.....	.....
1	15	15	15
1	15	15	9
.....	.....	.....	.....
1	3	3	2
1	3	2	2
.....	.....	.....	.....
2	3	8	8
2	8	8	8
.....	.....	.....	.....
2	29	29	29
2	29	29	31
.....	.....	.....	.....

(b) dataset_2 with two sliding window length.

Profil	P_0	P_1	P_2	P_3
.....	.....	.....	.....	.....
1	33	33	31	31
1	33	31	31	31
.....	.....	.....	.....	.....
1	3	3	2	2
1	3	2	2	2
.....	.....	.....	.....	.....
2	3	2	2	2
2	2	2	2	2
.....	.....	.....	.....	.....
2	28	28	27	27
2	28	27	27	27
.....	.....	.....	.....	.....

(c) dataset_3 with three sliding window length.

Profil	P_0	P_1	P_2	P_3	P_4
.....	.....	.....	.....	.....	.....
1	33	33	31	31	31
1	33	31	31	31	31
.....	.....	.....	.....	.....	.....
1	3	3	2	2	2
1	3	2	2	2	2
.....	.....	.....	.....	.....	.....
2	8	8	8	8	8
2	8	8	8	8	3
.....	.....	.....	.....	.....	.....
2	28	27	27	27	27
2	27	27	27	27	27
.....	.....	.....	.....	.....	.....

(d) dataset_4 with four sliding window length.

Profil	P_0	P_1	P_2	P_3	P_4	P_5
.....	.....	.....	.....	.....	.....	.....
1	33	33	31	31	31	31
1	33	31	31	31	31	31
.....	.....	.....	.....	.....	.....	.....
1	3	3	2	2	2	2
1	3	2	2	2	2	2
.....	.....	.....	.....	.....	.....	.....
2	8	8	8	8	3	3
2	8	8	8	3	3	3
.....	.....	.....	.....	.....	.....	.....
2	29	29	31	31	31	31
2	29	31	31	31	31	31
.....	.....	.....	.....	.....	.....	.....

(e) dataset_5 with five sliding window length.

Profil	P_0	P_1	P_2	P_3	P_4	P_5	P_6
.....	.....	.....	.....	.....	.....	.....	.....
1	33	33	31	31	31	31	31
1	33	31	31	31	31	31	31
.....	.....	.....	.....	.....	.....	.....	.....
1	3	3	2	2	2	2	2
1	3	2	2	2	2	2	2
.....	.....	.....	.....	.....	.....	.....	.....
2	8	8	8	3	3	3	3
2	8	8	3	3	3	3	3
.....	.....	.....	.....	.....	.....	.....	.....
2	29	31	31	31	31	31	31
2	31	31	31	31	31	31	29
.....	.....	.....	.....	.....	.....	.....	.....

(f) dataset_6 with six sliding window length.

Figure 5.3: Overall constructed datasets.

5.3 Experiment settings

In this section, we introduce the experimental setup, followed by experimental results, a comparison with different statistical and traditional machine learning models, and a discussion of these results. The experimental results in the following sections are presented within three stages: (1) We investigate the impacts of history length on the performance of the next-cell LSTM-based classifier; (2) We evaluate the performance of the proposed classifier using the different metrics often used in the multi-classification task; and (3) We conduct a comparative study with different baseline methods used in classification.

5.3.1 Datasets splitting for training and evaluation

To evaluate our LSTM-based model (and other models against which it is compared) in the three aforementioned stages, we have divided each dataset in the same following manner. We split each $\texttt{dataset}_k$, $k = (1, \cdots, 6)$ into two splits: training splits *D-train* and validation splits *D-test*. *D-train* split with 68% of dataset is used to train the model and *D-test* split with 32% of dataset is used to test and validate its performance.

The *D-train* split is formed by a juxtaposition of 68% of data of user's profile u_1 data with 68% of data of user's profile u_2. Similarly, the *D-test* split is formed by a juxtaposition of 32% of data of user's profile u_1 data with 32% of data of user's profile u_2.

In order to tune the hyper-parameters of the studied models, we have used Grid Search to find the best parameters for each model (see Table 5.1 for a summary of the best values for each model). Finally, we performed a prediction task on the whole test data *D-test*, and also on each of the 32% per-profile from the dataset.

5.3.2 Evaluation metrics

The performance metrics used to evaluate the LSTM model and the traditional machine learning models are : accuracy, precision, recall, and F1-Score [117, 118].

- **Accuracy:**
  - *Definition*: Accuracy is the ratio of correctly predicted instances to the total instances in the dataset.

  - *Formula:*

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

  - *Explanation:* This metric provides an overall effectiveness of the model by showing the proportion of true results (both true positives and true negatives) among the total number of cases examined.

- **Precision:**

  - *Definition*: Precision is the ratio of correctly predicted positive observations to the total predicted positives.

  - *Formula:*

$$\text{Precision} = \frac{TP}{TP + FP}$$

  - *Explanation:* Precision is useful for determining the exactness of the model. A high precision indicates that the model has a low false positive rate.

- **Recall:**

  - *Definition*: Recall, also known as sensitivity, is the ratio of correctly predicted positive observations to all the observations in the actual class.

  - *Formula:*

$$\text{Recall} = \frac{TP}{TP + FN}$$

  - *Explanation*: Recall is crucial for situations where it is important to capture all positive instances. A high recall indicates that the model has a low false negative rate.

- **F1-Score:**

  - *Definition*: The F1-Score is the harmonic mean of precision and recall.

  - *Formula:*

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

  - *Explanation:* The F1-Score is useful when seeking a balance between precision and recall, particularly in scenarios where there is an uneven class distribution.

Table 5.1: Hyper-parameter settings for each model

	Parameters	Best Parameters	Best Score
LSTM Model	Activation	elu	96.33%
	Batch Size	16	
	Epochs	50	
	Number of Neurons	64	
	Dropout Rate	0	
	Number of layers	3 LSTM, 1 Dense	
	Optimizer	ADAM	
	Loss Function	Categorical Cross_entropy	
	Classification Activation Function	Softmax	
	Parameters	Best Parameters	Best Score
KNN	Number of neighbors K	k=11	92.98
Random Forest	Number of trees K	n_estimators=10	95.99
	Maximum depth	15	
	Maximum number of leaf nodes	50	
SVM	C	C = 10	93.55
	kernel	rbf	
	gamma	scale	

5.3.3 Models hyper-parameters tuning

In order to select the best configuration for the studied models, we have used GridSearchCV on the proposed LSTM deep-learning estimator and the other studied models. We have applied the Grid Search technique over the parameter grid for each model by using the 5-fold cross-validation re-sampling method. The rationale for choosing the 5 folds is that we have a total 8573 number of entries (i.e., number of visited cells), and using more than 5 folds would decrease the fold's size and impact the validation task. The detailed configurations and hyper-parameters settings used for the proposed LSTM-based and other studied models

can be found in Table 5.1.

5.3.3.1 Hyperparameter selection for the LSTM model

For the LSTM RNN network, we have trained and tested the model in TensorFlow using Keras deep learning library [119]. We have selected a stacked-layer LSTM model with three layers as it gives better results compared to a single-layer LSTM. The number of epochs is fixed to 50, the optimizer and the loss function that were used are Adam and categorical_crossentropy, respectively. The output layer is a fully connected neural network with twenty-six units to specify the twenty-six candidate Cell IDs classes, followed by the Softmax activation function. While the following configurations of number of hidden cells, dropout rate, batch-size, and the activation function are explored with (32, 64, 128), (0, 0.1, 0.2), (10, 16, 32) and (relu, elu), respectively.

5.3.3.2 Hyperparameter selection of traditional ML models

As for the conventional ML algorithms, we have used the scikit-learn library to train and evaluate the generated models. For instance, to find the best fit K value for KNN algorithm, we have fixed the euclidean distance as a dissimilarity measure, while the values of the hyper-parameter k are varied from 1 to 500. Then, we run the Grid Search CV to pick the one with best cross-validation accuracy. As for the SVM algorithm, to find the best parameters that maximized the model's accuracy, we explored different values of hyperparameters, including the kernel type (rbf, sigmoid), regularization parameter C (1, 2, 10, 100), and gamma (scale, 1). For the RF algorithm, to identify the best hyperparameters of the model, we have used Grid Search CV with the following values: number of trees (10, 20, 30, 50, 60), maximum depth (15, 20, 25, 30), and minimum samples required at a leaf node (50, 100, 200, 250, 300, 400).

5.3.4 Results and analysis for each dataset

This section examines the impact of the history length on the performances of the future cell prediction. We performed a set of experiments on the proposed LSTM-based model, as well on baseline-methods used in classification such as KNN, RF, and SVM. Furthermore, for each experiment, we fixed the best configuration with 5-fold cross validation selected in

History length	Mean loss	Mean accuracy	Lowest loss	Highest accuracy
SW=1	21.52%	96.26%	17.25%	97.31%
SW=2	20.07%	96.12%	16.24%	97.31%
SW=3	22.09%	96.05%	19.11%	97.05%
SW=4	21.39%	95.82%	16.84%	97.12%
SW=5	21.74%	95.96%	18.32%	96.9%
SW=6	19.56%	96.34%	16.58%	97.27%

Table 5.2: LSTM model results with different sliding windows k

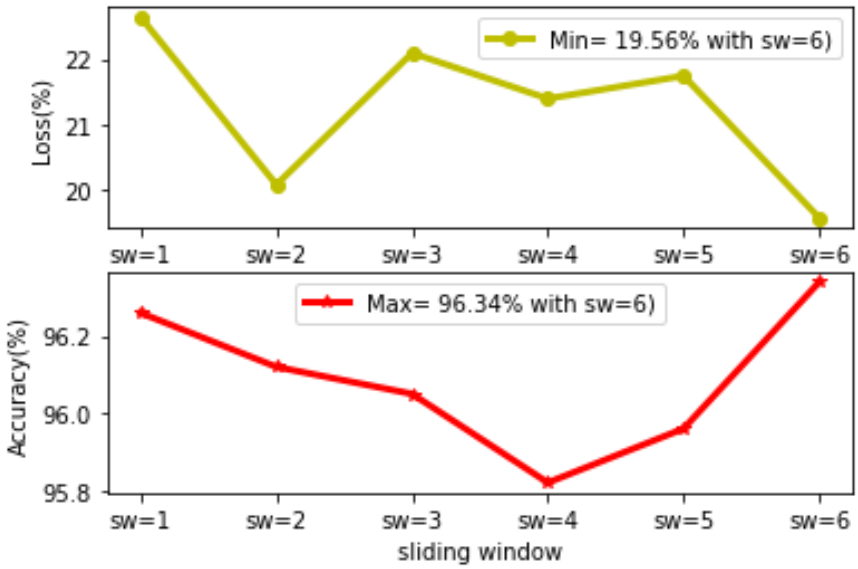

Figure 5.4: LSTM model loss and accuracy on test data versus history length.

Table 5.1. To train and validate each model, we used the training set *D-train* and the testing set *D-test* of each $\texttt{dataset}_k$, $k = (1, \cdots, 6)$. The prediction results on test data can be seen in Table 5.2 for LSTM model and Table 5.4 for traditional models. We note that the results of each experiment are obtained through simulations of 100 iterations.

5.3.4.1 Results of the LSTM classifier

Figure 5.4 shows the prediction loss and accuracy results for the LSTM model using the test data of each $\texttt{dataset}_k$. We analyzed the loss and accuracy metrics by varying the history length k (i.e., the sliding window), observing its effect on the model's performance. The numerical results with the maximum accuracy and lower loss for each dataset can be seen in Table 5.2. The obtained results, as shown in Figure 5.4, demonstrate good loss and accuracy values with LSTM model regardless of the history length. However, there are slight differences when the history length changes. Indeed, the loss results are varying from 22.09%

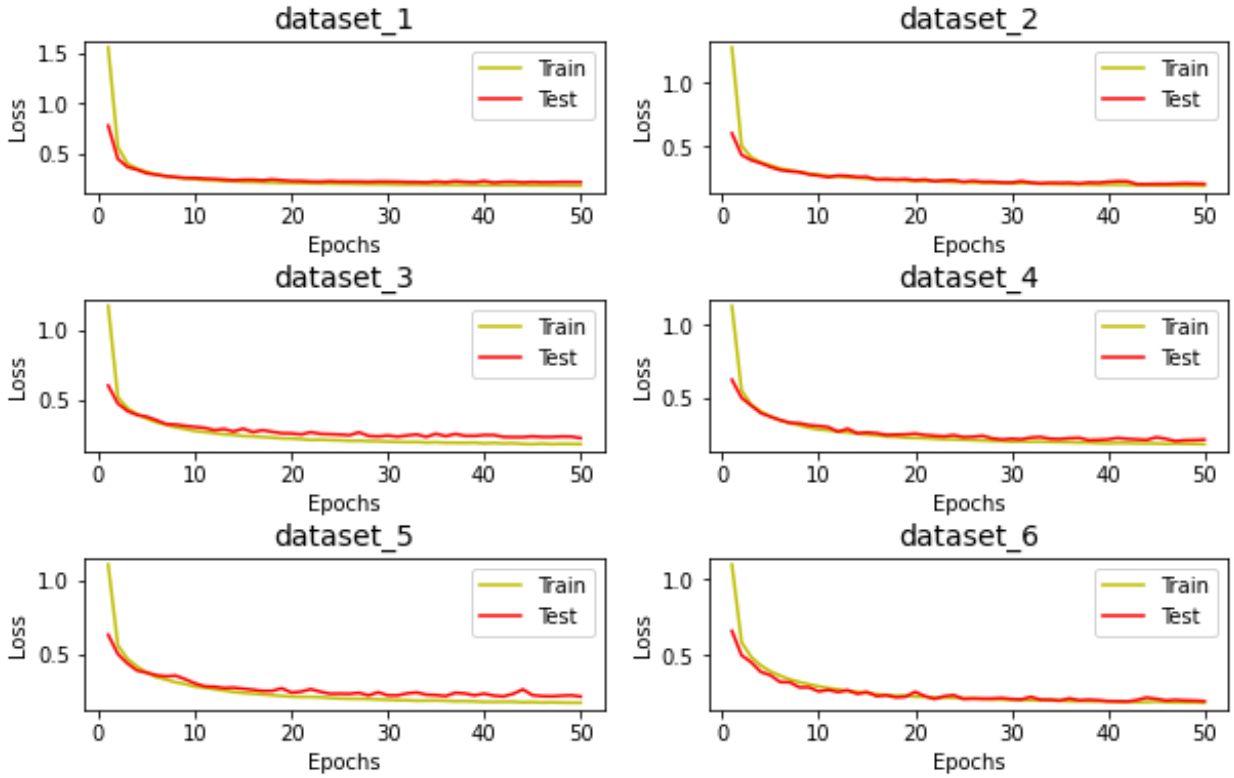

Figure 5.5: LSTM model loss on training and testing data for each $\texttt{dataset}_k$.

with $SW = 3$ to the lowest loss value of 19.56% with $SW = 6$, while the accuracy results are varying from the best result of 96.34% with $SW = 6$ to the worst result of 95.82% with $SW = 4$ (which is still acceptable). Finally, the proposed classifier brings better results for all datasets. However, it gives the best fit to the data of $\texttt{dataset}_6$ (i.e., with $SW = 6$) compared to other datasets, as it achieves the best accuracy and loss results of 96.34% and 19.56%, respectively.

To evaluate the performance of the LSTM-based predictions, we have also conducted a detailed comparison with the ground truth data (i.e., the training splits *D-train* of each $\texttt{dataset}_k$, $k = (1, \cdots, 6)$). Specifically, Figure 5.5 and 5.6 illustrate plots of the LSTM predictions alongside the ground truth data. The LSTM classifier was thus trained on the training splits *D-train* and validated using the testing splits *D-test* for each $\texttt{dataset}_k$, $(k = 1, \cdots, 6)$, with respect to loss and accuracy metrics over 50 training epochs. The results demonstrate a perfect convergence between the ground truth data and the predictions made by the model, thereby confirming the effectiveness and accuracy of the proposed LSTM-based mobility prediction framework. Furthermore, the quantitative analysis of the results, presented in Table 5.2, demonstrates about 21% loss and 96% accuracy regardless of the history length k. This reinforces our confidence in the robustness of the model and its

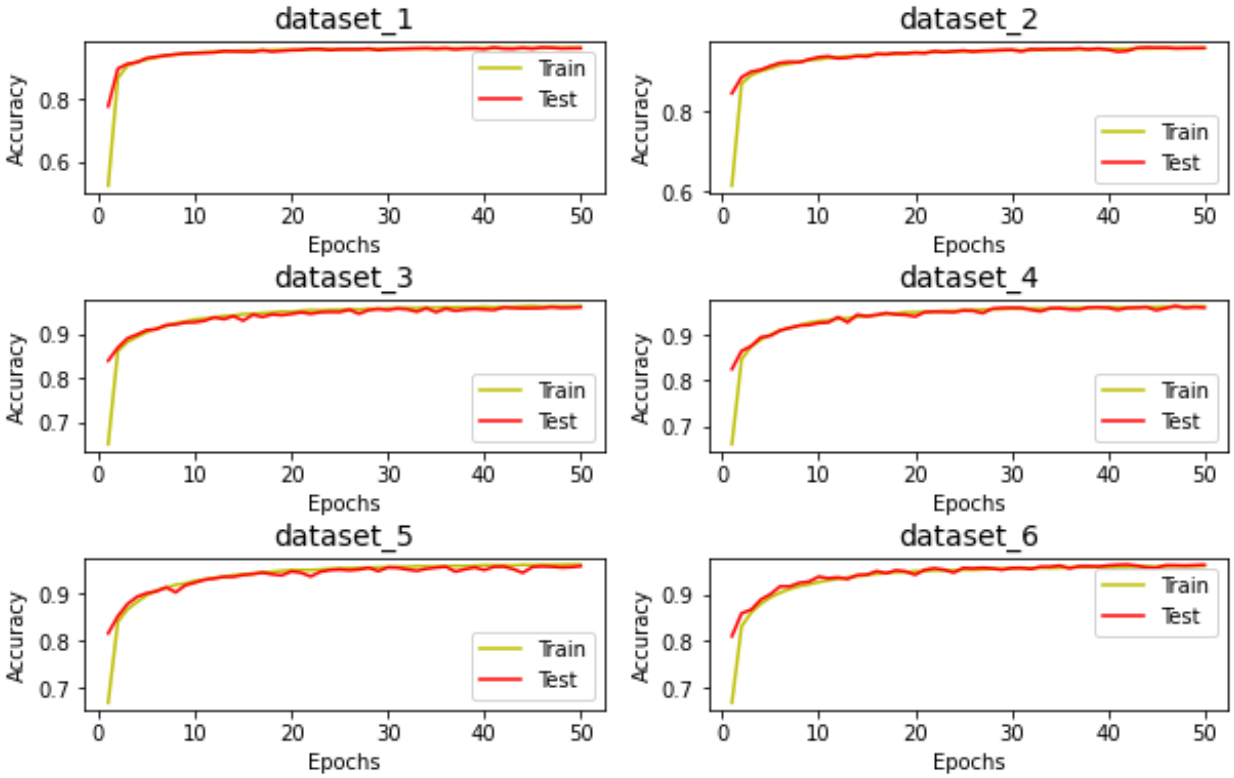

Figure 5.6: LSTM model accuracy on training and testing data for each $\texttt{dataset}_k$.

History length	SW=1	SW=2	SW=3	SW=4	SW=5	SW=6
Loss	18.87%	22.34%	27.25%	27.07%	28.89%	27.02%
Accuracy	97.34%	96.25%	94.68%	95.16%	94.86 %	94.42%

Table 5.3: KNN model results with different sliding windows k

ability to generalize effectively to new datasets without significant performance loss.

In the average of a set of experiments, the classifier can achieve better results of accuracy on training and testing data for each $\texttt{dataset}_k$. Figure 5.5 shows that the loss function is decreasing exponentially as the number of epochs increases. This indicates that the classifier gives better loss results on both training and testing data for each $\texttt{dataset}_k$. Similarly, Figure 5.6 shows that the accuracy metric is increasing exponentially as the number of epochs increases for train and test data at once.

5.3.4.2 Results of traditional ML classifiers

In this section, we examine the performance of machine learning models KNN, RF, and SVM for the next-cell prediction on both the training and testing sets. The experiments were conducted over an average of 100 model execution iterations.

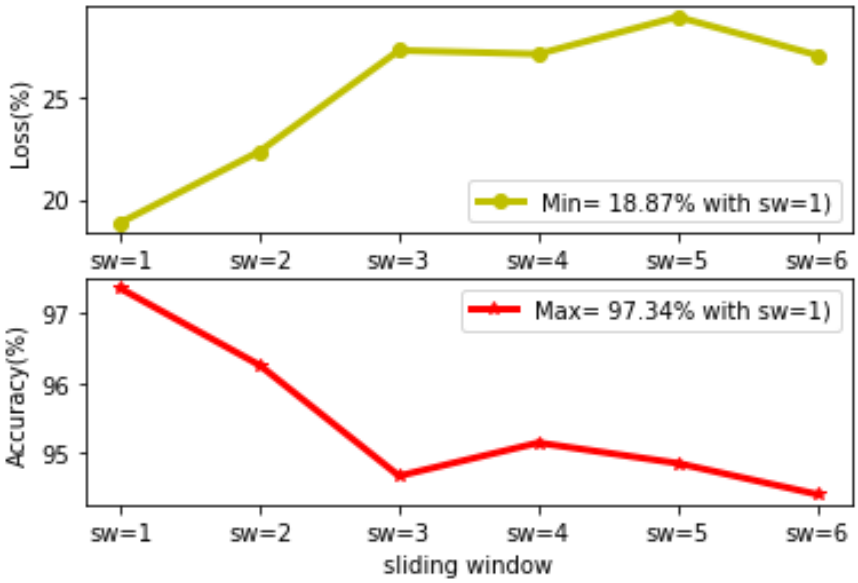

Figure 5.7: KNN model loss and accuracy on test data versus history length.

Results of the KNN model In the case of KNN, the prediction results are not influenced by the number of iterations as seen in Figure 5.8 and 5.9. This is because KNN is a deterministic algorithm, in which the prediction decision to classify each instance from the testing set is based on the distance measurement with the instances of the training set.

Meanwhile, the loss rate and accuracy results for the KNN model may be affected by the history length of each dataset as illustrated in Figure 5.7. We can see that the KNN algorithm performed good for each $\texttt{dataset}_k$ for the loss and the accuracy metrics, and

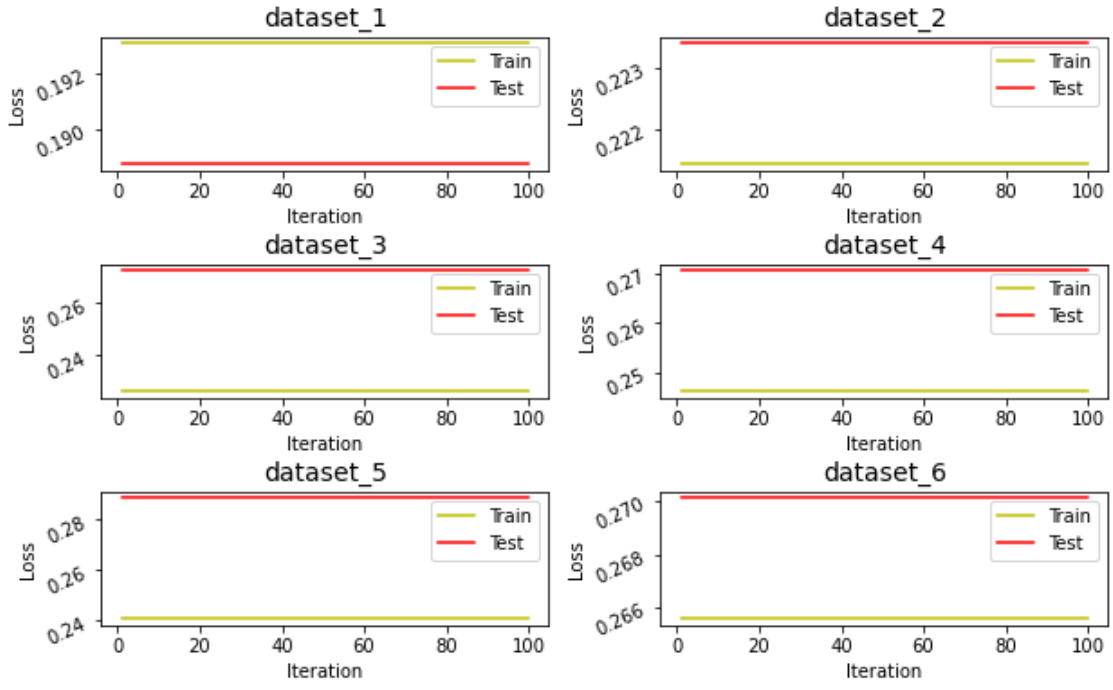

Figure 5.8: KNN model loss on training and testing data for each $\texttt{dataset}_k$.

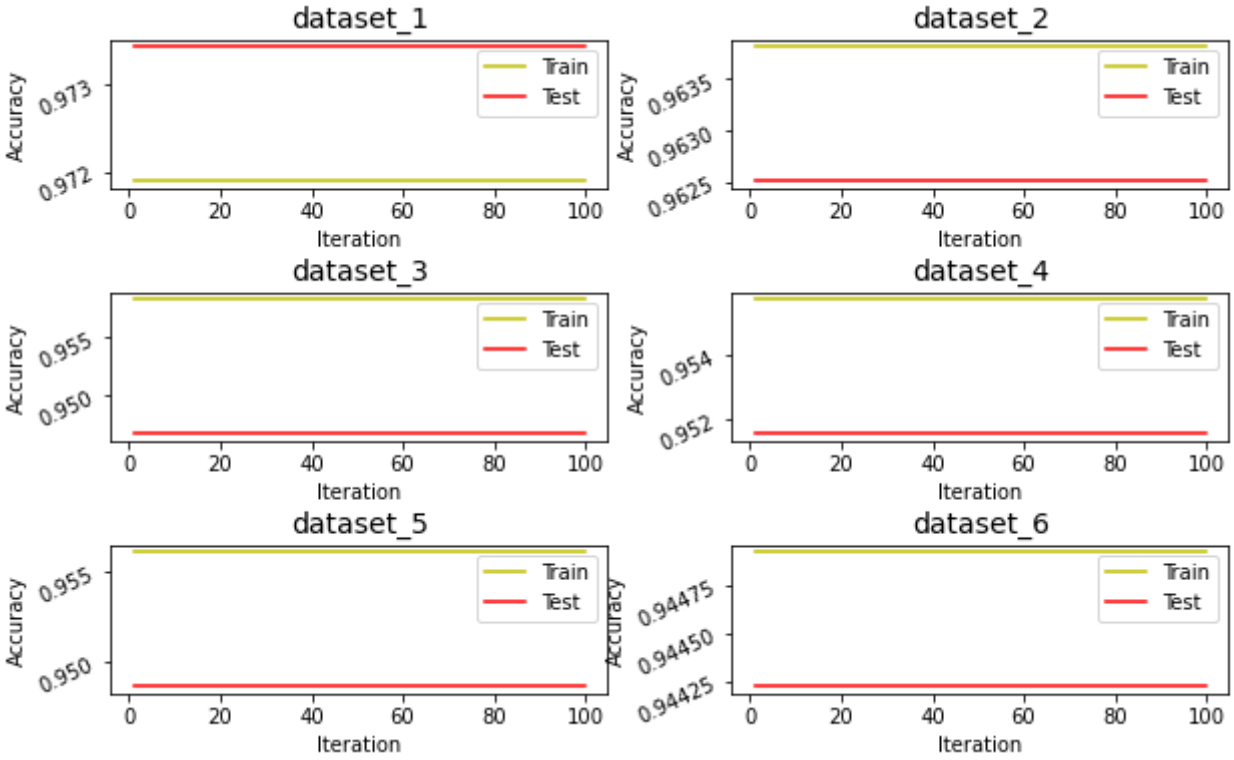

Figure 5.9: KNN model accuracy on training and testing data for each $\texttt{dataset}_k$.

performed best with the accuracy of 97.34% and the loss of 18.87% in the $\texttt{dataset}_1$, which has the lowest history length (i.e., with $SW = 1$). As it performed worst for the $\texttt{dataset}_6$ of highest history length (i.e., with $SW = 6$), with the accuracy of 94.42%, and the model achieved the highest loss on the $\texttt{dataset}_5$ (i.e., with $SW = 5$), where 28.89% of test data are misclassified. The performance evaluation of the KNN model for each dataset is shown in Table 5.3.

Results of RF model Similarly, the RF model may be influenced by the history length of each dataset. Figure 5.10 shows the loss and accuracy results for the RF model using the test data of each $\texttt{dataset}_k$. The RF model gives also best fitting to data of $\texttt{dataset}_1$ with

History length	Mean loss	Mean accuracy	Lowest loss	Highest accuracy
SW=1	37.91%	97.42%	36.71%	97.42%
SW=2	21.13%	97.25%	17.60 %	97.49%
SW=3	22.09%	96.32%	19.24 %	96.72%
SW=4	19.65%	96.00%	17.12 %	96.54 %
SW=5	19.63%	96.13%	17.98 %	96.72%
SW=6	18.55%	95.81%	16.76 %	96.97%

Table 5.4: RF model results with different sliding windows k

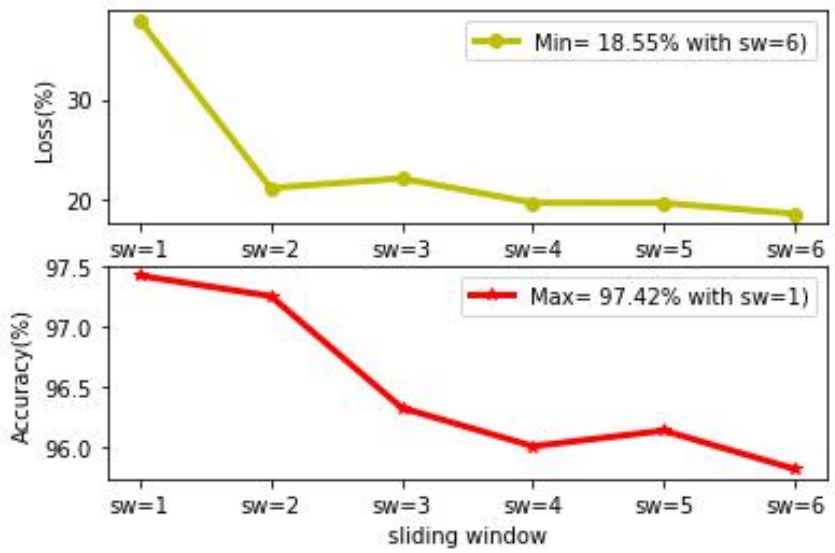

Figure 5.10: RF model loss and accuracy on test data versus history length.

accuracy of 97.42%, and achieved best loss on the dataset $\texttt{dataset}_5$ with proportional of 18.55% of data are wrongly classified. The numerical results showing the highest accuracy and lowest loss for each dataset are presented in Table 5.4.

However, the RF model may be influenced by the number of iterations this is due to the inherent randomness of the model. The correlation between training and testing curves

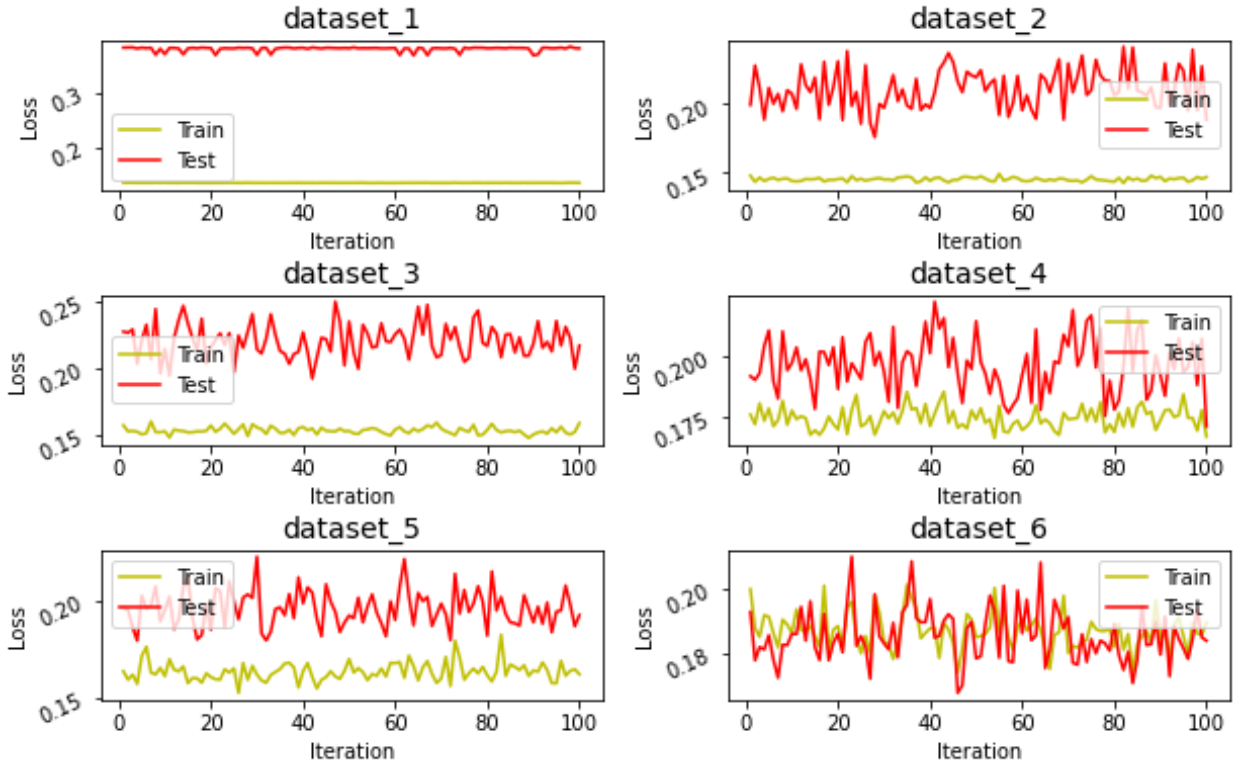

Figure 5.11: RF model loss on training and testing data for each $\texttt{dataset}_k$.

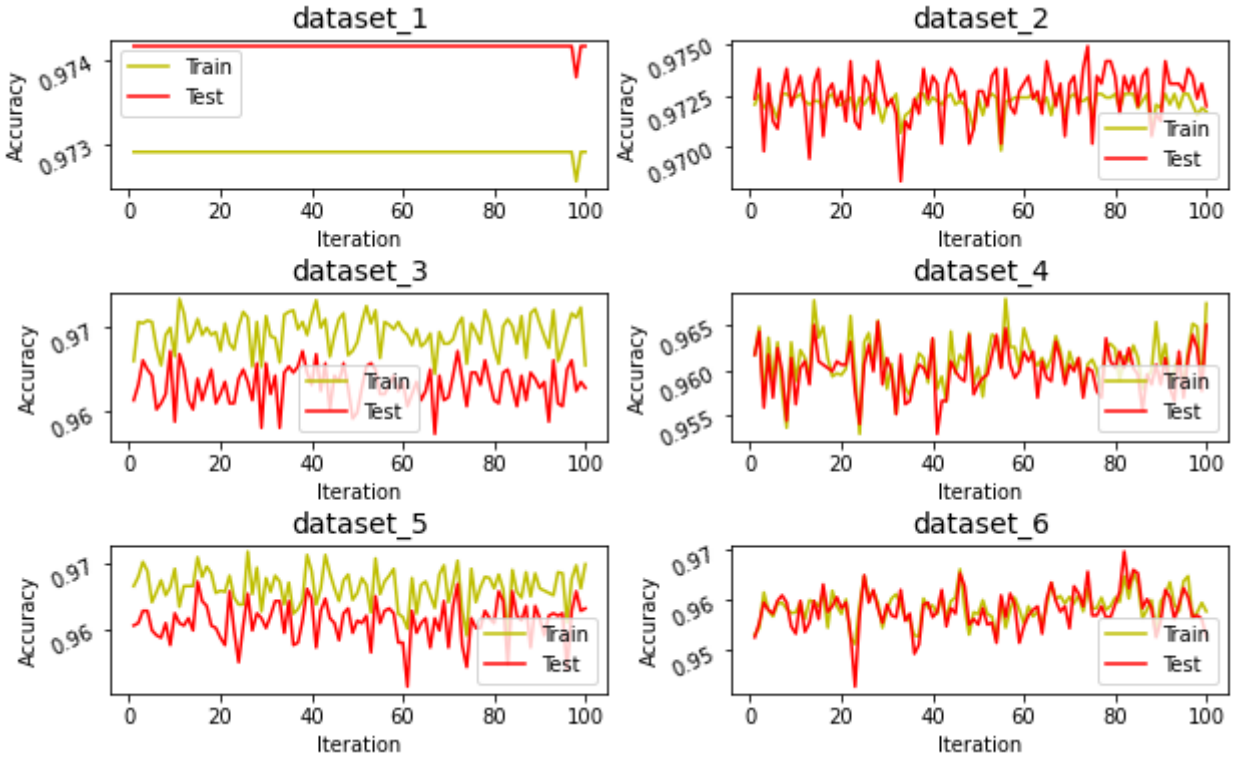

Figure 5.12: RF model accuracy on training and testing data for each $\texttt{dataset}_k$.

History length	Mean loss	Mean accuracy	Lowest loss	Highest accuracy
SW=1	21.85%	95.52%	20.68%	95.52%
SW=2	22.01%	95.09%	21.07%	95.09%
SW=3	23.73%	94.65%	22.68%	94.65%
SW=4	24.18%	94.32%	23.18%	94.32%
SW=5	23.82%	93.95%	23.18%	93.95%
SW=6	22.92%	94.42%	22.21%	94.42%

Table 5.5: SVM model results with different sliding windows k

for the next-cell prediction using the RF model is illustrated in Figure 5.11 and 5.12. It shows the loss and the accuracy results over 100 iterations of training, and indicates that the performance of the RF classifier is well satisfying for all datasets. However, in such curves, we observed that the validation curve is slightly outperforming the training curve. For the loss curves, this is found on all dataset except for the $\texttt{dataset}_7$. In contrary, in the accuracy curves, the RF model performed well for all datasets except for $\texttt{dataset}_1$, where we observe that the training and the testing curves are a little uncorrelated.

Results of SVM model In the case of SVM model, Figure 5.13 shows the loss and the accuracy results for the SVM model using the test data of each $\texttt{dataset}_k$. The SVM model preformed well on $\texttt{dataset}_1$ with an accuracy of 95.52%, and a loss rate of 21.85%.

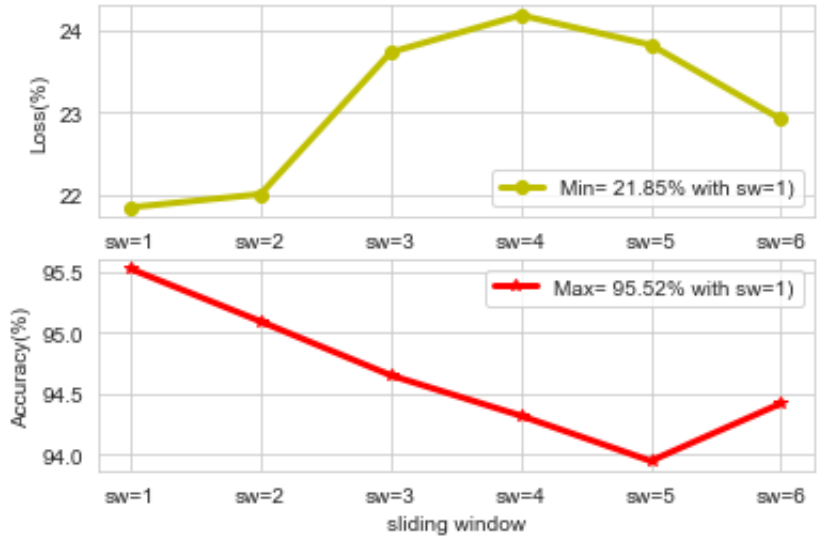

Figure 5.13: SVM model loss and accuracy on test data versus history length.

Table 5.4 shows the numerical results of the highest accuracy and the lowest loss for each dataset. The training and testing curves of each dataset, depicted in Figure 5.14 and 5.15, show that the SVM classifier performs effectively across all datasets. However, we observe that the validation curve slightly outperforms the training curve in certain datasets. This is found on $\texttt{dataset}_6$ and $\texttt{dataset}_4$ of the accuracy curves, and in $\texttt{dataset}_3$, $\texttt{dataset}_4$, and $\texttt{dataset}_5$ of the loss curves.

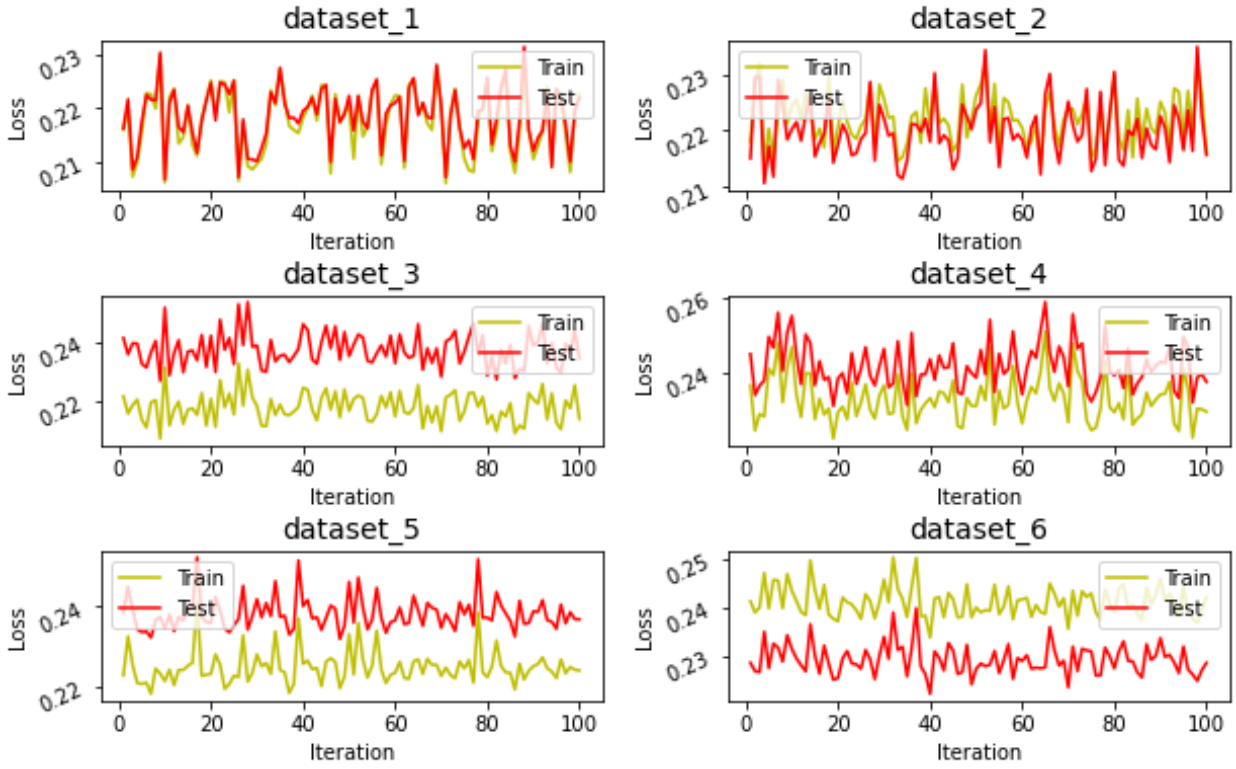

Figure 5.14: SVM model loss on training and testing data for each $\texttt{dataset}_k$.

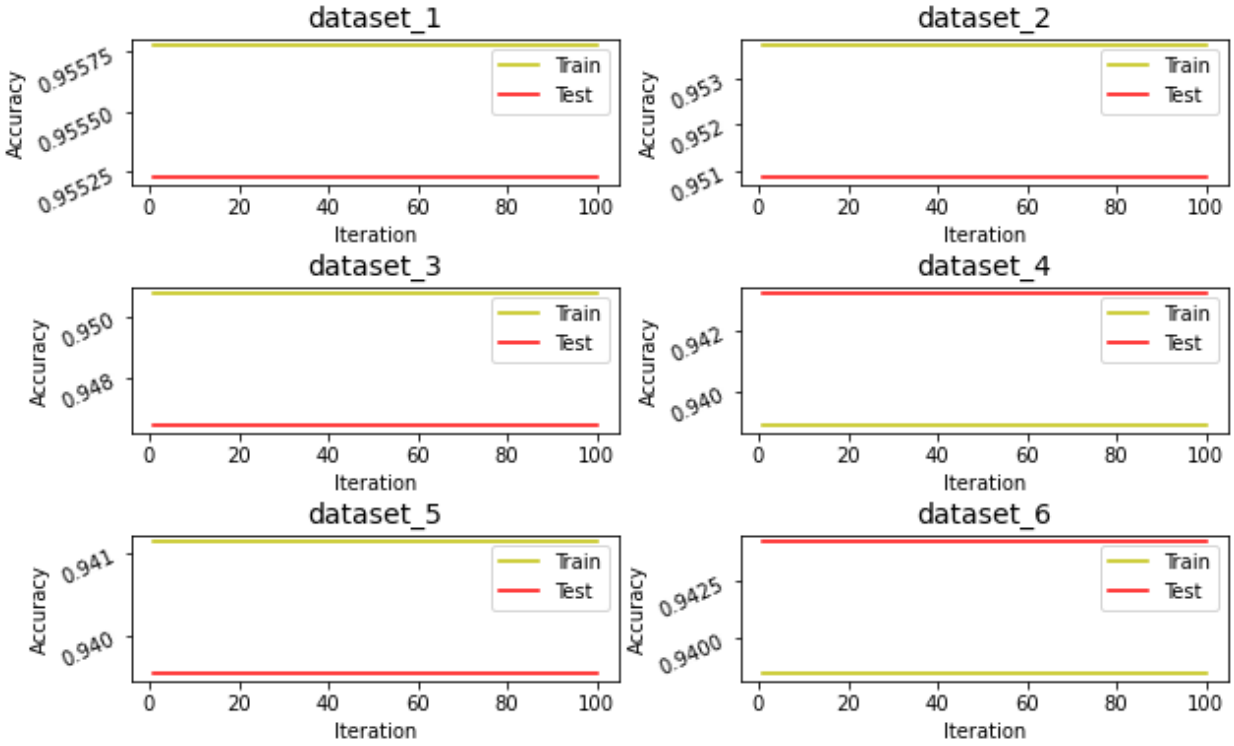

Figure 5.15: SVM model accuracy on training and testing data for each $\texttt{dataset}_k$.

5.3.5 Results and analysis using multi-classification metrics

Herein, we started the experiments on $\texttt{dataset}_6$ to evaluate the performance of the proposed LSTM classifier as well with three traditional ML models, namely KNN, RF, and SVM. To train each model, we set the parameters that have been tuned before in Table 5.1 by using the training set *D-train*. Then, we evaluate the prediction results for each model by using the different metrics used in the classification task.

5.3.5.1 Model evaluation using confusion matrix

Figure 5.16 shows the overall confusion matrix accuracy of the next-cell LSTM-based prediction model. The LSTM classifier gives 2662 true predicted cells from 2743 real cells of test data, thus a result of about 97.05% accuracy score, as shown in Table 5.6. The CM accuracy evaluation per mobility profile is also conducted. The LSTM classifier performed better with user's profile u_2 with accuracy of 98.05%, where 1614 cells are well predicted among 1646 true cells of test data of user's profile u_2. While, in user's profile u_1, the accuracy can reach a 95.53% score with 1048 true predicted cells among 1097 real cells of test data of user's profile u_1.

In the case of KNN algorithm, the overall CM of the model is shown in Figure 5.17. We

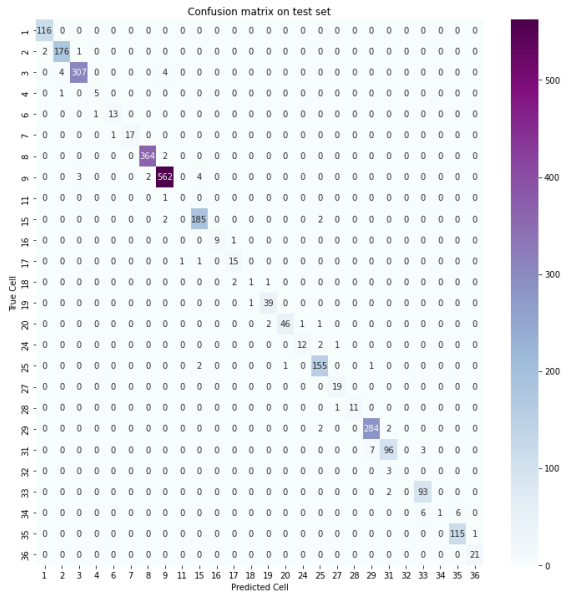

Figure 5.16: LSTM Confusion Matrix.

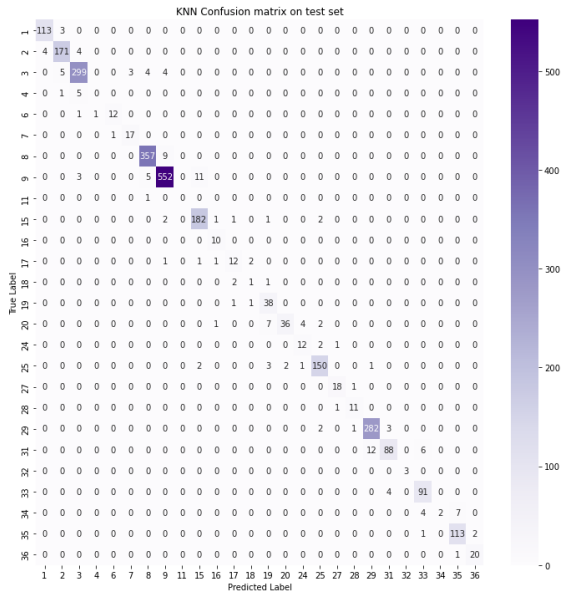

Figure 5.17: KNN Confusion Matrix.

obtained an accuracy of about 94.42%, with 1016 true predicted cells well-classified among 2743 ground truth data of the test data. As more, the KNN model provides higher accuracy on user's profile u_1 in comparison to user's profile u_2. The classifier demonstrated a correct prediction of 1016 cells from 1097 true labels of data test, and 1574 truly predicted cells among 1646 true labels of testing data, with a percentage of 95.63% of user's profile u_2 data and 92.62% of user's profile u_1 data accurately classified.

Concerning the RF classifier, the overall CM of the model can be found in Figure 5.18. The best accuracy can be reached at about 95.77% with 2627 predicted cells correctly classified among 2627 real test data labels. For user's profile u_1, the accuracy can achieve about 94.26% with 1034 of predicted cells truly classified from 1097 of test data. As for user's profile u_2, the accuracy may give about 96.78% with 1593 well predicted cells from 1646 of test data. The classifier performed as same as other models with the user's profile u_2 data.

Similarly, the overall CM of the SVM model is depicted in Figure 5.19. The SVM classifier predict 2590 true predicted cells from 2743 real labels of test data, thus a result of about 94.42% accuracy performance. The accuracy for each mobility profile gives about 94.26% of data for user's profile u_1 with 1034 correctly predicted cells from the testing data, and about 94.53% with 1556 true predicted cells from 1097 real cells of test data. The classifier achieved nearly equal accuracy for each user's profile.

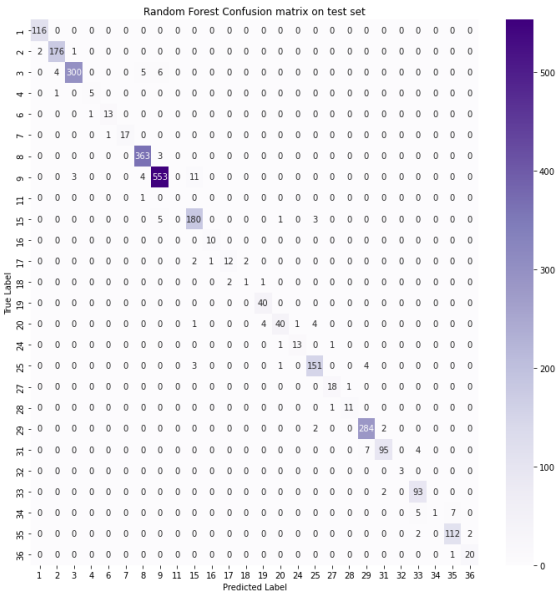

Figure 5.18: RF Confusion Matrix.

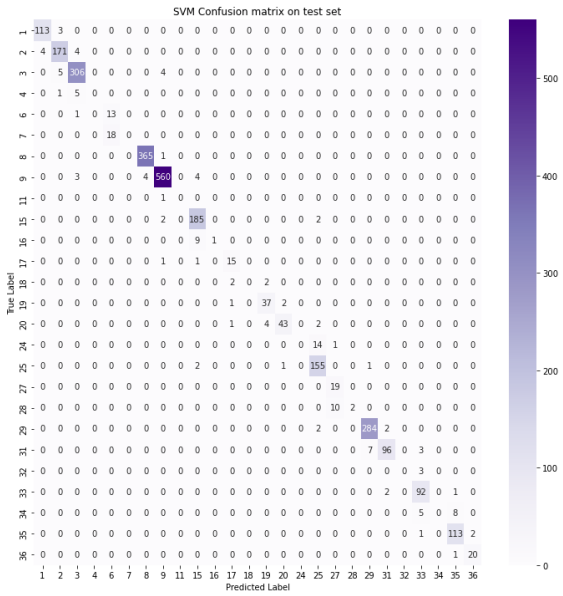

Figure 5.19: SVM Confusion Matrix.

Model	Accuracy%	f1_score%	Precision%	Recall%
LSTM	97.05%	96.79%	96.95%	97.05%
KNN	94.42%	94.17%	94.36%	94.42%
SVM	94.42%	93.30%	92.93%	94.42%
Random Forest	95.77%	95.54%	95.78%	95.77%

Table 5.6: Performance of LSTM classifier and the traditional ML classifiers

5.3.5.2 Model evaluation using F1_Score, precision , and recall metrics

Table 5.6 and Figure 5.20 show the superiority of the LSTM model compared to the traditional methods in term of averaged accuracy, F1_Score, precision, and recall metrics. As more, the LSTM-based next-cell prediction model achieved a precision of 96.95%, a recall of 97.05%, and a F1_Score of 96.79% for the averaged number of classes. Followed by the RF model, which brings an average precision of 95.78%, a recall of 95.77%, and a F1_Score of 95.54% for the averaged number of classes. Then, the KNN algorithm reported an average F1-Score of around 94.17%, a precision of 94.36%, and a recall of 94.42%. Finally, the SVM reported the lowest results with an average precision of 92.93%, a recall of 94.42% and a F1_Score of 93.30%.

5.3.5.3 Discussion

As outlined by the experiments that have been conducted for the evaluation of the proposed LSTM-based next-cell prediction framework, the key finding of the current research is that

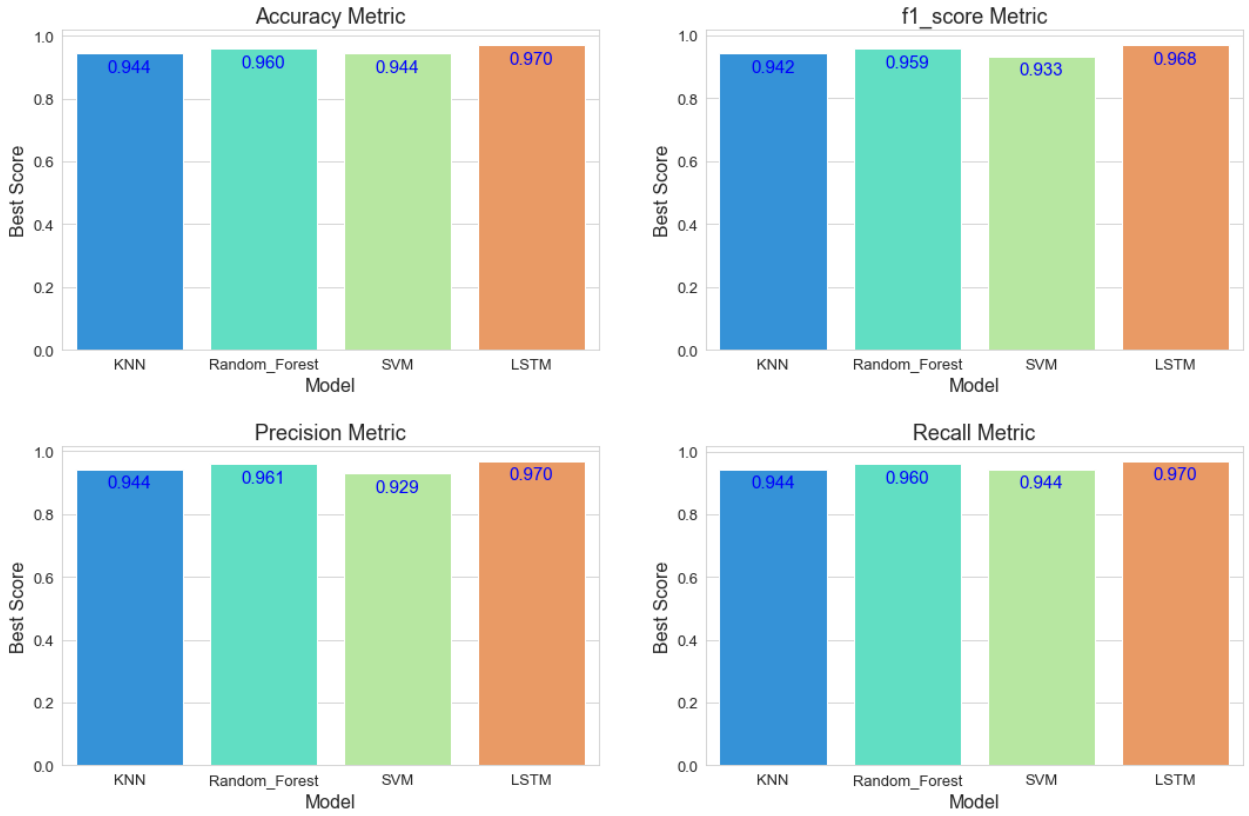

Figure 5.20: Comparison of the LSTM classifier with other ML methods.

an LSTM model is performing better than the conventional models in vehicle mobility data. This is because the LSTM is designed to capture long-term temporal dependencies through specialized architecture characterized by internal memory cells and gates. As demonstrated by our results experiments, the effectiveness of LSTM classifier in the $\texttt{dataset}_6$ outperforms other datasets with lower history length, as same it exceeds the performance of the baseline classifiers KNN, RF, and SVM for the next-cell prediction. As more, in the average of 100 experiments, the LSTM classifier achieved the best accuracy and loss of 96.34% and 19.56%, respectively on $\texttt{dataset}_6$. Against of the KNN classifier which brings lowest results on $\texttt{dataset}_6$, with accuracy of 94.42% and loss of 27.02%. As same as the RF model which performed the worst results in $\texttt{dataset}_6$ with loss of 18.55% and an accuracy of 95.81%. While the SVM model achieved the lowest results in $\texttt{dataset}_6$ with an accuracy of 94.42% and a loss of 22.92%. In contrast, all ML models performed well in $\texttt{dataset}_1$, and with better accuracy than the LSTM model , but the performance of each model is a little degraded as the history length increased unlike the LSTM model. This is due to the lack the ability of ML models to capture time dependencies from sequential data or time series data.

In the second stage based on the conducted experiments on $\texttt{dataset}_6$, we evaluated

Model	$u_1\%$	$u_2\%$
LSTM	$95,53$	$98,05$
KNN	$92,62$	$95,63$
SVM	94.26	$94,53$
RF	$94,26$	$96,78$

Table 5.7: Accuracy per profile

the performance of the LSTM classifier using the different metrics often used in the multi-classification task. We started by the confusion matrix, among all models, LSTM was found to perform better than KNN, RF, and SVM as LSTM was able to capture sequence information better then traditional ML models. Regarding the obtained results, LSTM classifier brings better accuracy of 97.05% followed by RF with 95.77%, then KNN and SVM performed the same accuracy of 94.42%. The accuracy obtained for each mobility profile were also taken into account (see Table 5.7), where each model provides higher accuracy on user's mobility profile u_2 versus user's profile u_1, against the SVM model which attained almost the same accuracy across each profile. This is due to the fact that user's profile u_2 represents $3/5$ of the training and testing sets. As more the LSTM model reported average F1-Score, precision, and recall better than all the baseline models, followed by the RF model, then the KKN model, and finally the SVM which performed worst than other models. At the end, the proposed LSTM-based model for the next-cell prediction consistently outperforms baseline methods when dealing with large time-series sequences, showing better accuracy, precision, recall, and F1-Score results. This demonstrates that LSTM is more adept at understanding and forecasting temporal patterns in the data compared to baseline methods. LSTM has the capability to capture the mobility patterns even from complex datasets with multiple users and a variety of features, which profile users' daily style.

5.4 Practical integration of the proposed approach

5.4.1 Scalability concerns

The scalability of our LSTM model is crucial for real-world 5G mc-IoT deployments. Our model is designed to handle extensive volumes of vehicle mobility data efficiently. The architecture leverages historical mobility data and processes it in real-time, which allows it to scale as the number of connected devices increases. Our experiments demonstrate that

the model maintains high accuracy and performance. This scalability is achieved through efficient data management techniques and the ability to predict next-cell transitions with minimal latency, ensuring that the system can accommodate the dynamic nature of 5G networks without degradation in service quality.

5.4.2 Computational requirements

A significant advantage of our LSTM-based approach is its low computational requirements. The LSTM architecture is specifically designed to handle long-term dependencies in sequential data, which reduces the need for extensive computational resources. By focusing on the historical mobility data of vehicles and employing a multi-class classification approach, our model minimizes computational overhead. This efficiency is further supported by optimized data preprocessing techniques and a streamlined model structure, allowing the system to function on standard hardware without requiring specialized, high-performance computing resources. Our experimental results indicate that the model can process data quickly and accurately, making it suitable for real-time applications in dynamic environments with prediction times of less than 1 ms, as demonstrated in the Figure 5.21. The table in the figure is a snapshot of the results, showing the average prediction time per sample over 100 iterations using the CPU. Each sequence is chosen randomly, with both the predicted and true next cells displayed. Only one sample was misclassified, with an average execution time of $0.78ms$.

5.4.3 Potential challenges

Implementing the LSTM-based approach in real-world settings involves several challenges, such as managing data transmission latency, ensuring model robustness in diverse environments, and maintaining data privacy. To address these issues, we conducted extensive testing using realistic mobility scenarios. Our model demonstrated the ability to adapt to varying mobility conditions, ensuring reliable performance while maintaining data privacy and security. The LSTM model's robustness and adaptability make it a viable solution for enhancing the performance and reliability of 5G mc-IoT systems.

	Index	Predicted_cell	True_cell	Prediction_Time
0	272	2	2	0.738900
1	31	1	1	0.695336
2	1683	8	8	0.857191
3	1844	9	9	0.674629
4	1971	8	8	0.668528
...	...	...	...	...
95	1439	9	9	0.685768
96	2109	9	9	0.702848
97	576	35	35	0.852902
98	292	9	9	0.699761
99	438	3	3	0.673418

Figure 5.21: Prediction Execution Time per one_sequence.

5.5 Conclusion

This chapter evaluated the the effectiveness of the proposed LSTM classifier in high-mobility, mission-critical use cases. Our simulation experiments were performed using realistic mobility traces in a vehicular environment case study. The real vehicle mobility dataset was obtained from SUMO. We performed a series of experiments on the classifier using datasets with various history lengths, and the results have validated the effectiveness of the performed predictions on short-term mobility prediction. Additionally, the results showed that the proposed classifier performs better on longer history datasets. While compared to traditional Machine Learning algorithms used for classification, the proposed LSTM model outperformed ML methods, achieving the highest prediction accuracy.

Chapter 6

Conclusions and Future Works

This thesis centered on the mission critical communication scenarios IoT applications which is intended for time-sensitive applications that need high speed, high reliability, and low latency. Throughout this academic work, to introduce this mcMTC communication slice group provided by 5G networks. We delved into the foundational concepts of the IoT and the 5G networks, and the associated technologies shaping the future evolution of IoT, including softwarisation, virtualization and, in detail,network slicing techniques.

To ensure the end-to-end delivery of mission-critical services with QoS, it is essential to explore IoT systems from the standpoint of 5G cellular networks. 5G networks are considered the first generation of mobile networks specifically designed to meet the diverse requirements of IoT systems and various industry verticals. Therefore, we began by examined the telecommunication systems from the inception of the first generation, providing an overview of the existing cellular network generations. Then we highlighted the various stages of 5G mobile networks development, citing the key advancements introduced by 5G that have facilitated the adoption of IoT systems. Further, we have presented the core softwarisation and virtualization technologies designed to provide 5G cellular network with greater flexibility, programmability, and adaptability. These advantages have enabled large-scale IoT deployment, addressing critical challenges such as device diversity, mobility, heterogeneity, security, and scalability.

We have studied the concept of mobility within the 5G infrastructure and explored how mobility prediction can significantly enhance predictive maintenance strategies. A state-of-the-art review of mobility prediction methods in 5G communication systems is elaborated, where we have classified, surveyed and compared them. In this context, we have presented

a taxonomy of mobility prediction solutions used in 5G, classifying them into two main categories, namely : history-based and measurement-driven strategies. We then categorized the most relevant contributions identified in the literature, illustrating the critical role of mobility prediction in predictive maintenance, with a particular focus on efficient traffic management, dynamic resource allocation, and seamless network slicing. Finally, we conducted a comprehensive comparative analysis of the reviewed works, evaluating the various mobility prediction methods discussed.

The research contribution of this ~~dissertation~~ thesis lies in the proposal of a mobility prediction architecture based on LSTM for the prediction of the next cell. This architecture serves as a predictive maintenance framework within 5G-based network slicing systems, in which the network be able to determine in which cell the user is located in near real-time for proactive service management. Hence, this facilitates preemptive resource allocation, and seamless mobility and handover support for network slices. Therefore, the core benefit of the proposed approach is the effective management of mission-critical IoT slices for time-sensitive and high-mobility applications, by reducing latency and improving the reliability of mc-IoT communications.

Though we have discussed the mobility prediction algorithm using the LSTM framework, which is designed to further optimize network efficiency and reliability under dynamic conditions targeting mc-IoT network slices. Several open issues remain to be explored in the future, such as :

- **LSTM model improvement:** We aim to enhance our LSTM model to carry out long-term mobility predictions across heterogeneous mobility profiles like train, drone and so on. Additionally, we plan to extend into a multi-user prediction framework with multi-variant features, integrating several other mobility-influencing factors such as user velocity, received signal strength, channel state information, and others. Because network slicing architecture originated for diversified application scenarios, and vehicular use cases are one example type that supports different slice classes, we aim to further improve our model while taking into account the different slice types (i.e., mMTC, eMBB, and URLLC).

- **A deep integration within the predictive maintenance process:** As 5G evolves into an heterogeneous environment with diverse high-mobility use cases, predictive

maintenance must advance accordingly. This involves leveraging AI, advanced analytics, and real-time slicing to anticipate and prevent potential failures or performance degradation in virtualized network slices, highlighting the critical role of mobility prediction in predictive maintenance. Our future work will involve implementing the developed model in a practical 5G communication system, where the predicted cell information can be leveraged to automate critical slice control and management. This will help optimize resource allocation and enhance mobility management for these slices. Similarly, it is essential for 5G network slicing control and management to prioritize mcMTC slice updates over eMBB and mMTC to meet their stringent reliability and latency requirements. This synergy drives 5G toward self-optimizing networks that proactively detect and mitigate disruptions before impacting performance.

Bibliography

[1] Ivan Farris, Tarik Taleb, Yacine Khettab, and Jaeseung Song. A survey on emerging sdn and nfv security mechanisms for iot systems. *IEEE Communications Surveys Tutorials*, 21(1):812–837, 2019.

[2] Rashid Mijumbi, Joan Serrat, Juan-Luis Gorricho, Niels Bouten, Filip De Turck, and Raouf Boutaba. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys Tutorials*, 18(1):236–262, 2016.

[3] ETSI membership. Network functions virtualisation (nfv); architectural framework. 2014. Retrieved octobre 16, 2024, from `http://www.etsi.org`.

[4] Sherif Abdelwahab, Bechir Hamdaoui, Mohsen Guizani, and Taieb Znati. Network function virtualization in 5g. *IEEE Communications Magazine*, 54(4):84–91, 2016.

[5] Sihem Bakri. *Towards enforcing network slicing in 5G networks*. PhD thesis, Sorbonne Université, 2021.

[6] Chong Han, Josep Miquel Jornet, Etimad Fadel, and Ian F Akyildiz. A cross-layer communication module for the internet of things. *Computer Networks*, 57(3):622–633, 2013.

[7] Mamta Agiwal, Abhishek Roy, and Navrati Saxena. Next generation 5g wireless networks: A comprehensive survey. *IEEE communications surveys & tutorials*, 18(3):1617–1655, 2016.

[8] 3GPP. 5g; service requirements for the 5g system. 2021.

[9] Xuemin Shen, Jie Gao, Wen Wu, Kangjia Lyu, Mushu Li, Weihua Zhuang, Xu Li, and Jaya Rao. Ai-assisted network-slicing based next-generation wireless networks. *IEEE Open Journal of Vehicular Technology*, 1:45–66, 2020.

[10] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. elib Electronic Library, German Aerospace Center (DLR), November 2018.

[11] Luis Felilpe Ariza Vesga. Many-objective problems optimization focused on energy efficiency applied to 5g heterogeneous cellular networks using the small cell switch-off framework. 2020.

[12] Nandish P Kuruvatti. *Incorporating Context Awareness in Cellular Networks to Enhance System Performance and User Mobility Support*. PhD thesis, Technische Universität Kaiserslautern, 2020.

[13] Amitabha Ghosh, Andreas Maeder, Matthew Baker, and Devaki Chandramouli. 5g evolution: A view on 5g cellular technology beyond 3gpp release 15. *IEEE access*, 7:127639–127651, 2019.

[14] 3GPP. The 3gpp complete work plan. `https://www.3gpp.org/ftp/Information/WORK_PLAN/`, 2024. Accessed: 2024-10-22.

[15] Xingqin Lin. An overview of 5g advanced evolution in 3gpp release 18. *IEEE Communications Standards Magazine*, 6(3):77–83, 2022.

[16] Qualcomm Incorporated. What's next in 5g advanced? `https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/5G-A-Rel-19-Presentation.pdf`, 2024. Accessed: 2024-10-22.

[17] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines. 5g network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges. *Computer Networks*, 167:106984, 2020.

[18] Diego Kreutz, Fernando M. V. Ramos, Paulo Esteves Veríssimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015.

[19] TR-521. Sdn architecture. `https://opennetworking.org/wp-content/uploads/2014/10/TR-521_SDN_Architecture_issue_1.1.pdf`, 2016. Accessed: 2024-10-22.

[20] Matteo Pozza, Patrick K Nicholson, Diego F Lugones, Ashwin Rao, Hannu Flinck, and Sasu Tarkoma. On reconfiguring 5g network slices. *IEEE Journal on Selected Areas in Communications*, 38(7):1542–1554, 2020.

[21] Fabrizio Granelli, Anteneh A Gebremariam, Muhammad Usman, Filippo Cugini, Veroniki Stamati, Marios Alitska, and Periklis Chatzimisios. Software defined and virtualized wireless access in future wireless networks: scenarios and standards. *IEEE Communications Magazine*, 53(6):26–34, 2015.

[22] Bo Han, Vijay Gopalakrishnan, Lusheng Ji, and Seungjoon Lee. Network function virtualization: Challenges and opportunities for innovations. *IEEE communications magazine*, 53(2):90–97, 2015.

[23] Khizar Abbas, Talha Ahmed Khan, Muhammad Afaq, and Wang-Cheol Song. Network slice lifecycle management for 5g mobile networks: An intent-based networking approach. *IEEE Access*, 9:80128–80146, 2021.

[24] Ali Esmaeily and Katina Kralevska. Small-scale 5g testbeds for network slicing deployment: A systematic review. *Wireless Communications and Mobile Computing*, 2021(1):6655216, 2021.

[25] Xueli An, Chan Zhou, Riccardo Trivisonno, Riccardo Guerzoni, Alexandros Kaloxylos, David Soldani, and Artur Hecker. On end to end network slicing for 5g communication systems. *Transactions on Emerging Telecommunications Technologies*, 28(4):e3058, 2017.

[26] Min Chen, Yiming Miao, Hamid Gharavi, Long Hu, and Iztok Humar. Intelligent traffic adaptive resource allocation for edge computing-based 5g networks. *IEEE Transactions on Cognitive Communications and Networking*, 6(2):499–508, 2020.

[27] Y Series. Network slice orchestration and management for providing network services to 3rd party in the imt-2020 network. *Recommendation ITU-T*, 3153, 2019. Retrieved August 17, 2022, from `https://www.itu.int/rec/T-REC-Y.3153/`.

[28] document ITU-M Series Mobile. Imt vision – framework and overall objectives of the future development of imt for 2020 and beyond. 2015.

[29] Fumihiro Hasegawa, Akinori Taira, Gosan Noh, Bing Hui, Hiroshi Nishimoto, Akihiro Okazaki, Atsushi Okamura, Junhwan Lee, and Ilgyu Kim. High-speed train communications standardization in 3gpp 5g nr. *IEEE Communications Standards Magazine*, 2(1):44–52, 2018.

[30] Nurul Huda Mahmood, Hirley Alves, Onel Alcaraz López, Mohammad Shehab, Diana P Moya Osorio, and Matti Latva-Aho. Six key features of machine type communication in 6g. In *2020 2nd 6G Wireless Summit (6G SUMMIT)*, pages 1–5, Levi, Finland, 2020. IEEE.

[31] Carsten Bockelmann, Nuno Pratas, Hosein Nikopour, Kelvin Au, Tommy Svensson, Cedomir Stefanovic, Petar Popovski, and Armin Dekorsy. Massive machine-type communications in 5g: Physical and mac-layer solutions. *IEEE communications magazine*, 54(9):59–65, 2016.

[32] Qi Zhang and Frank HP Fitzek. Mission critical iot communication in 5g. In *Future Access Enablers for Ubiquitous and Intelligent Infrastructures: First International Conference, FABULOUS 2015, Ohrid, Republic of Macedonia, September 23-25, 2015. Revised Selected Papers 1*, pages 35–41. Springer, 2015.

[33] Arjun Anand, Gustavo De Veciana, and Sanjay Shakkottai. Joint scheduling of urllc and embb traffic in 5g wireless networks. *IEEE/ACM Transactions On Networking*, 28(2):477–490, 2020.

[34] M Series. Minimum requirements related to technical performance for imt-2020 radio interface (s). *Report*, 2410:2410–2017, 2017.

[35] Ema Becirovic. *On Massive MIMO for Massive Machine-Type Communications*. PhD thesis, Linköping University Electronic Press, 2020.

[36] Nurul Huda Mahmood, Stefan Böcker, Andrea Munari, Federico Clazzer, Ingrid Moerman, Konstantin Mikhaylov, Onel Lopez, Ok-Sun Park, Eric Mercier, Hannes Bartz, et al. White paper on critical and massive machine type communication towards 6g. *arXiv preprint arXiv:2004.14146*, page arXiv:2004.14146, 2020.

[37] Mohd Muntjir, Mohd Rahul, and Hesham A Alhumyani. An analysis of internet of

things (iot): novel architectures, modern applications, security aspects and future scope with latest case studies. *Int. J. Eng. Res. Technol*, 6(6):422–447, 2017.

[38] Kevin Ashton et al. That 'internet of things' thing. *RFID journal*, 22(7):97–114, 2009.

[39] Frederic Thiesse and Florian Michahelles. An overview of epc technology. *Sensor review*, 26(2):101–105, 2006.

[40] document ITU Internet Reports. The internet of things. *Report*, 2005.

[41] Riya Sil and Ritwesh Chatterjee. *Evolution of Next-Generation Communication Technology*, pages 1–17. Springer Nature Singapore, Singapore, 2023.

[42] Kinza Shafique, Bilal A Khawaja, Farah Sabir, Sameer Qazi, and Muhammad Mustaqim. Internet of things (iot) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5g-iot scenarios. *Ieee Access*, 8:23022–23040, 2020.

[43] Roberto Minerva, Abyi Biru, and Domenico Rotondi. Towards a definition of the internet of things (iot). *IEEE Internet Initiative*, 1(1):1–86, 2015.

[44] Ovidiu Vermesan and Peter Friess. *Internet of things: converging technologies for smart environments and integrated ecosystems*. River publishers, 2013.

[45] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, 17(4):2347–2376, 2015.

[46] Mahyar Shirvanimoghaddam, Massimo Condoluci, Mischa Dohler, and Sarah J Johnson. On the fundamental limits of random non-orthogonal multiple access in cellular massive iot. *IEEE Journal on Selected Areas in Communications*, 35(10):2238–2252, 2017.

[47] Samad Ali, Nandana Rajatheva, and Walid Saad. Fast uplink grant for machine type communications: Challenges and opportunities. *IEEE Communications Magazine*, 57(3):97–103, 2019.

[48] Antonino Orsino, Aleksandr Ometov, Gabor Fodor, Dmitri Moltchanov, Leonardo Militano, Sergey Andreev, Osman N. C. Yilmaz, Tuomas Tirronen, Johan Torsner, Giuseppe Araniti, Antonio Iera, Mischa Dohler, and Yevgeni Koucheryavy. Effects of heterogeneous mobility on d2d- and drone-assisted mission-critical mtc in 5g. *IEEE Communications Magazine*, 55(2):79–87, 2017.

[49] Afif Osseiran, Federico Boccardi, Volker Braun, Katsutoshi Kusume, Patrick Marsch, Michal Maternia, Olav Queseth, Malte Schellmann, Hans Schotten, Hidekazu Taoka, et al. Scenarios for 5g mobile and wireless communications: the vision of the metis project. *IEEE communications magazine*, 52(5):26–35, 2014.

[50] Zhidan Liu, Zhenjiang Li, Kaishun Wu, and Mo Li. Urban traffic prediction from mobility data using deep learning. *IEEE Network*, 32(4):40–46, 2018.

[51] Fa Li, Zhipeng Gui, Zhaoyu Zhang, Dehua Peng, Siyu Tian, Kunxiaojia Yuan, Yunzeng Sun, Huayi Wu, Jianya Gong, and Yichen Lei. A hierarchical temporal attention-based lstm encoder-decoder model for individual mobility prediction. *Neurocomputing*, 403:153–166, 2020.

[52] Roshan Fernandes and Rio D'Souza GL. A new approach to predict user mobility using semantic analysis and machine learning. *Journal of medical systems*, 41(12):1–12, 2017.

[53] Zhanwei Hou, Changyang She, Yonghui Li, Li Zhuo, and Branka Vucetic. Prediction and communication co-design for ultra-reliable and low-latency communications. *IEEE Transactions on Wireless Communications*, 19(2):1196–1209, 2019.

[54] Ke Xiao, Jianyu Zhao, Yunhua He, and Shui Yu. Trajectory prediction of uav in smart city using recurrent neural networks. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pages 1–6, Shanghai, China, 2019. IEEE.

[55] Nasrin Bahra and Samuel Pierre. A hybrid user mobility prediction approach for handover management in mobile networks. 2(2):199–212, 2021.

[56] Umberto Fattore, Marco Liebsch, Bouziane Brik, and Adlen Ksentini. Automec: Lstm-based user mobility prediction for service management in distributed mec resources. In *MSWiM '20: Proceedings of the 23rd International ACM Conference on Modeling,*

Analysis and Simulation of Wireless and Mobile Systems, pages 155–159, Alicante, Spain, 2020. ACM.

[57] Katja Gilly, Sonja Filiposka, and Salvador Alcaraz. Predictive migration performance in vehicular edge computing environments. *Applied Sciences*, 11(3):944, 2021.

[58] Wei Liu and Yozo Shoji. Edge-assisted vehicle mobility prediction to support v2x communications. *IEEE Transactions on Vehicular Technology*, 68(10):10227–10238, 2019.

[59] Seong Hyeon Park, ByeongDo Kim, Chang Mook Kang, Chung Choo Chung, and Jun Won Choi. Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1672–1678, Changshu, China, 2018. IEEE.

[60] Nandish P. Kuruvatti, Harold Moses Mutabazi, and Hans D. Schotten. Exploiting mobility context awareness in cellular networks for assisting vehicular use cases. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–7, Chicago, IL, USA, 2018. IEEE.

[61] Nandish P. Kuruvatti, Sachinkumar Bavikatti Mallikarjun, Sai Charan Kusumapani, and Hans D. Schotten. Mobility awareness in cellular networks to support service continuity in vehicular users. In *2020 3rd International Conference on Information and Communications Technology (ICOIACT)*, pages 431–435, Yogyakarta, Indonesia, 2020. IEEE.

[62] Amar Sinha, Venkanna Uduthalapally, Debanjan Das, and Rajarshi Mahapatra. Sdn-based seamless mobility management for b5g services in high-speed railways. In *2023 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 294–299, 2023.

[63] Suchao Xiao and Wen Chen. Dynamic allocation of 5g transport network slice bandwidth based on lstm traffic prediction. In *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, pages 735–739, Beijing, China, 2018. IEEE.

[64] Zhipeng Cheng, Ning Chen, Bang Liu, Zhibin Gao, Lianfen Huang, Xiaojiang Du, and Mohsen Guizani. Joint user association and resource allocation in hetnets based on user mobility prediction. *Computer Networks*, 177:107312, 2020.

[65] Munazza Shabbir, Sithamparanathan Kandeepan, Akram Al-Hourani, and Wayne Rowe. Lstm based proactive access point selection and mobility load balancing for ultra-dense networks. In *2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 452–458, 2024.

[66] Changyang She, Yifan Duan, Guodong Zhao, Tony QS Quek, Yonghui Li, and Branka Vucetic. Cross-layer design for mission-critical iot in mobile edge computing systems. *IEEE Internet of Things Journal*, 6(6):9360–9374, 2019.

[67] Rongpeng Li, Chujie Wang, Zhifeng Zhao, Rongbin Guo, and Honggang Zhang. The lstm-based advantage actor-critic learning for resource management in network slicing with user mobility. *IEEE Communications Letters*, 24(9):2005–2009, 2020.

[68] Muhammad Ashar Tariq, Malik Muhammad Saad, Mahnoor Ajmal, Ayesha Siddiqa, Junho Seo, Yang Haishan, and Dongkyun Kim. Network slice traffic demand prediction for slice mobility management. In *2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 281–285, 2024.

[69] Asma Belhadj, Karim Akilal, Siham Bouchelaghem, Mawloud Omar, and Sofiane Aissani. Next-cell prediction with lstm based on vehicle mobility for 5g mc-iot slices. *Telecommunication Systems*, 87(3):809–833, 2024.

[70] Murtaza Ahmed Siddiqi, Heejung Yu, and Jingon Joung. 5g ultra-reliable low-latency communication implementation challenges and operational issues with iot devices. *Electronics*, 8(9):981, 2019.

[71] 3GPP. 5g; service requirements for enhanced v2x scenarios(3gpp ts 22.186 version 16.2.0 release 16). 2020. Retrieved March 16, 2022, from `https://www.etsi.org/deliver`.

[72] He Chen, Rana Abbas, Peng Cheng, Mahyar Shirvanimoghaddam, Wibowo Hardjawana, Wei Bao, Yonghui Li, and Branka Vucetic. Ultra-reliable low latency cellular

networks: Use cases, challenges and approaches. *IEEE Communications Magazine*, 56(12):119–125, 2018.

[73] Michele Luvisotto, Zhibo Pang, and Dacfey Dzung. Ultra high performance wireless control for critical applications: Challenges and directions. *IEEE Transactions on Industrial Informatics*, 13(3):1448–1459, 2017.

[74] Bernd Holfeld, Dennis Wieruch, Thomas Wirth, Lars Thiele, Shehzad Ali Ashraf, Jorg Huschke, Ismet Aktas, and Junaid Ansari. Wireless communication for factory automation: An opportunity for lte and 5g systems. *IEEE Communications Magazine*, 54(6):36–43, 2016.

[75] Malla Reddy Sama, Sergio Beker, Wolfgang Kiess, and Srisakul Thakolsri. Service-based slice selection function for 5g. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2016.

[76] Najib Ahmed Mohammed, Ali Mohammed Mansoor, and Rodina Binti Ahmad. Mission-critical machine-type communication: An overview and perspectives towards 5g. *IEEE Access*, 7:127198–127216, 2019.

[77] Gàbor Fodor, Julia Vinogradova, Peter Hammarberg, Keerthi Kumar Nagalapur, Zhiqiang Tyler Qi, Hieu Do, Ricardo Blasco, and Mirza Uzair Baig. 5g new radio for automotive, rail, and air transport. *IEEE Communications Magazine*, 59(7):22–28, 2021.

[78] Jingxian Wu and Pingzhi Fan. A survey on high mobility wireless communications: Challenges, opportunities and solutions. *IEEE Access*, 4:450–476, 2016.

[79] Zhiqiang Xiao and Yong Zeng. An overview on integrated localization and communication towards 6g. *Science China Information Sciences*, 65(3):1–46, 2022.

[80] Zhanwei Hou, Changyang She, Yonghui Li, Dusit Niyato, Mischa Dohler, and Branka Vucetic. Intelligent communications for tactile internet in 6g: Requirements, technologies, and challenges. *IEEE Communications Magazine*, 59(12):82–88, 2021.

[81] Haijun Zhang, Na Liu, Xiaoli Chu, Keping Long, Abdol-Hamid Aghvami, and Victor

C. M. Leung. Network slicing based 5g and future mobile networks: Mobility, resource management, and challenges. *IEEE communications magazine*, 55(8):138–145, 2017.

[82] Vinod Kumar Choyi, Ayman Abdel-Hamid, Yogendra Shah, Samir Ferdi, and Alec Brusilovsky. Network slice selection, assignment and routing within 5g networks. In *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 1–7. IEEE, 2016.

[83] Satish Kumar, Rahul Banerji, Naman Gupta, Suman Kumar, Sukhdeep Singh, Avinash Bhat, Seungil Yoon, and Shatarupa Dash. Mas5g: Move around smartly in 5g. In *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, pages 214–221. IEEE, 2019.

[84] Spyridon Vassilaras, Lazaros Gkatzikis, Nikolaos Liakopoulos, Ioannis N Stiakogiannakis, Meiyu Qi, Lei Shi, Liu Liu, Merouane Debbah, and Georgios S Paschos. The algorithmic aspects of network slicing. *IEEE Communications Magazine*, 55(8):112–119, 2017.

[85] Ved P. Kafle, Yusuke Fukushima, Pedro Martinez-Julia, and Takaya Miyazawa. Consideration on automation of 5g network slicing with machine learning. In *2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*, pages 1–8. IEEE, 2018.

[86] Jinhe Zhou, Wenjun Zhao, and Shuo Chen. Dynamic network slice scaling assisted by prediction in 5g network. *IEEE Access*, 8:133700–133712, 2020.

[87] Anurag Thantharate, Rahul Paropkari, Vijay Walunj, and Cory Beard. Deepslice: A deep learning approach towards an efficient and reliable network slicing in 5g networks. In *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 0762–0767. IEEE, 2019.

[88] Linyu Huang, Liang Lu, and Wei Hua. A survey on next-cell prediction in cellular networks: Schemes and applications. *IEEE Access*, 8:201468–201485, 2020.

[89] Changyang She, Chengjian Sun, Zhouyou Gu, Yonghui Li, Chenyang Yang, H Vincent Poor, and Branka Vucetic. A tutorial on ultrareliable and low-latency communications in 6g: Integrating domain knowledge into deep learning. *Proceedings of the IEEE*, 109(3):204–246, 2021.

[90] Luke B Godfrey and Michael S Gashler. Neural decomposition of time-series data for effective generalization. *IEEE Transactions on Neural Networks and Learning Systems*, 29(7):2973–2985, 2017.

[91] Yuxiu Hua, Zhifeng Zhao, Rongpeng Li, Xianfu Chen, Zhiming Liu, and Honggang Zhang. Deep learning with long short-term memory for time series prediction. *IEEE Communications Magazine*, 57(6):114–119, 2019.

[92] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning(ICML '08)*, pages 160–167, New York, United States, 2008. ACM.

[93] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In Eric P. Xing and Tony Jebara, editors, *International conference on machine learning*, volume 32, pages 1764–1772, Bejing, China, 2014. PMLR. Retrieved from `https://proceedings.mlr.press/v32/graves14.html`.

[94] Changqing Luo, Jinlong Ji, Qianlong Wang, Xuhui Chen, and Pan Li. Channel state information prediction for 5g wireless communications: A deep learning approach. *IEEE Transactions on Network Science and Engineering*, 7(1):227–236, 2018.

[95] Adita Kulkarni, Anand Seetharam, Arti Ramesh, and J Dinal Herath. Deepchannel: Wireless channel quality prediction using deep learning. *IEEE Transactions on Vehicular Technology*, 69(1):443–456, 2019.

[96] Xueshi Hou, Jianzhong Zhang, Madhukar Budagavi, and Sujit Dey. Head and body motion prediction to enable mobile vr experiences with low latency. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7, Waikoloa, HI, USA, 2019. IEEE.

[97] Wenjing Zhang, Yuan Liu, Tingting Liu, and Chenyang Yang. Trajectory prediction with recurrent neural networks for predictive resource allocation. In *2018 14th IEEE International Conference on Signal Processing (ICSP)*, pages 634–639, Beijing, China, 2018. IEEE.

[98] Gary White, Andrei Palade, and Siobhán Clarke. Forecasting qos attributes using lstm networks. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, Rio de Janeiro, Brazil, 2018. IEEE.

[99] Stefania Bartoletti, Andrea Conti, Davide Dardari, and Andrea Giorgetti. 5g localization and context-awareness. In *5G Italy White Book: From Research to Market*, pages 167–187. National, Inter-Univ. Consortium for Telecommunications, University of Bologna; University of Ferrara, Italy, 2018.

[100] Jukka Talvitie, Toni Levanen, Mike Koivisto, Tero Ihalainen, Kari Pajukoski, and Mikko Valkama. Positioning and location-aware communications for modern railways with 5g new radio. *IEEE Communications Magazine*, 57(9):24–30, 2019.

[101] Nicola Bui, Matteo Cesana, S Amir Hosseini, Qi Liao, Ilaria Malanchini, and Joerg Widmer. A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques. *IEEE Communications Surveys & Tutorials*, 19(3):1790–1821, 2017.

[102] Paulo Valente Klaine, Muhammad Ali Imran, Oluwakayode Onireti, and Richard Demo Souza. A survey of machine learning techniques applied to self-organizing cellular networks. *IEEE Communications Surveys & Tutorials*, 19(4):2392–2431, 2017.

[103] Rahul Arun Paropkari, Anurag Thantharate, and Cory Beard. Deep-mobility: A deep learning approach for an efficient and reliable 5g handover. In *2022 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pages 244–250, Chennai, India, 2022.

[104] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[105] Abdelhadi Azzouni and Guy Pujolle. A long short-term memory recurrent neural network framework for network traffic matrix prediction. *arXiv preprint arXiv:1705.05690*, 2017.

[106] Daoud Burghal, Ashwin T Ravi, Varun Rao, Abdullah A Alghafis, and Andreas F Molisch. A comprehensive survey of machine learning based localization with wireless signals. *arXiv preprint arXiv:2012.11171*, 2020.

[107] Christopher Olah. Understanding lstm networks, August 2015. Retrieved August 18, 2022, from `https://colah.github.io/posts/2015-08-Understanding-LSTMs/`.

[108] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[109] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[110] Zhenghua Chen, Han Zou, JianFei Yang, Hao Jiang, and Lihua Xie. Wifi fingerprinting indoor localization using local feature-based deep lstm. *IEEE Systems Journal*, 14(2):3001–3010, 2019.

[111] Hasim Sak, Andrew W. Senior, and Françoise Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH*, pages 338–342, 2014.

[112] Urminder Singh, Sucheta Chauhan, A Krishnamachari, and Lovekesh Vig. Ensemble of deep long short term memory networks for labelling origin of replication sequences. In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–7, Paris, France, 2015. IEEE.

[113] Claude E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[114] P.J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[115] Siham Bouchelaghem. Mobility prediction, 2021. figshare `https://github.com/siham-bouchelaghem/mobility-prediction`.

[116] OpenStreetMap Contributors. Openstreetmap. `https://www.openstreetmap.org`, 2015.

[117] Amir Basati and Mohammad Mehdi Faghih. Efficient iot network intrusion detection using deep feature extraction. *Telecommunication Systems*, 34(18):15175–15195, 2022.

[118] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

[119] Francois Chollet et al. Keras github. `https://github.com/fchollet/keras`, 2015.

Abstract:

The evolution of 5G networks introduces dynamic, virtualized environments where seamless connectivity and quality of service are essential. Mobility management is crucial for optimizing resource allocation, ensuring service continuity, and enhancing predictive maintenance. Machine-type communications support diverse IoT applications requiring ultra-reliable, low-latency communication. Network slicing addresses these needs by differentiating between mission-critical and massive communication slices. This thesis explores the role of mobility prediction in improving the quality of service by forecasting user transitions between network cells, enabling proactive service management. A next-cell prediction framework using Long Short-Term Memory networks is proposed. Evaluated in vehicular networks, the model outperforms conventional classifiers, demonstrating its potential to enhance predictive maintenance and resource allocation.

Keywords: 5G, Network slicing, Predictive maintenance, QoS, LSTM.

Résumé :

L'évolution des réseaux 5G introduit des environnements dynamiques et virtualisés où la connectivité transparente et la qualité de service sont essentielles. La gestion de la mobilité est cruciale pour optimiser l'allocation des ressources, assurer la continuité du service et améliorer la maintenance prédictive. Les communications de type machine prennent en charge diverses applications IoT nécessitant une communication ultra-fiable et à faible latence. Le découpage réseau répond à ces besoins en distinguant les tranches de communication critique et massive. Cette thèse explore le rôle de la prédiction de mobilité dans l'amélioration de la qualité de service en anticipant les transitions des utilisateurs entre les cellules réseau, permettant une gestion proactive des services. Un cadre de prédiction de la prochaine cellule basé sur les réseaux de mémoire à long terme est proposé. Évalué dans des réseaux véhiculaires, le modèle surpasse les classificateurs conventionnels, démontrant son potentiel à améliorer la maintenance prédictive, anticiper la dégradation des performances, et faciliter l'allocation des ressources.

Mots-clés : 5G, Découpage réseau, Maintenance prédictive, QdS, LSTM.

ملخص

قدم تطور شبكات الجيل الخامس بيئات ديناميكية وافتراضية حيث يعد الاتصال السلس وجودة الخدمة أمرين أساسيين. تلعب إدارة التنقل دورًا حاسمًا في تحسين تخصيص الموارد، وضمان استمرارية الخدمة، وتعزيز الصيانة التنبؤية. تدعم اتصالات الأجهزة المتنوعة تطبيقات إنترنت الأشياء التي تتطلب اتصالًا فائق الموثوقية ومنخفض الكمون. يعالج تقسيم الشبكة هذه الاحتياجات من خلال التمييز بين شرائح الاتصالات الحرجة والضخمة. تستكشف هذه الأطروحة دور التنبؤ بالتنقل في تحسين جودة الخدمة من خلال توقع انتقالات المستخدمين بين الخلايا الشبكية، مما يتيح إدارة استباقية للخدمات. نقترح إطارًا للتنبؤ بالخلايا التالية يعتمد على شبكات الذاكرة طويلة المدى (لصةپ). تم تقييم النموذج في شبكات المركبات، وأظهر تفوقًا على المصنفات التقليدية، مما يثبت قدرته على تحسين الصيانة التنبؤية وتخصيص الموارد.

الكلمات المفتاحية : الجيل الخامس $5G$ ، تقسيم الشبكة، الصيانة التنبؤية، جودة الخدمة QoS ، شبكات الذاكرة طويلة المدى $LSTM$.